

日 本 国 特 許 庁
JAPAN PATENT OFFICE

31000 U.S. PRO
10/075989
02/15/02

別紙添付の書類に記載されている事項は下記の出願書類に記載されて
いる事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed
with this Office

出 願 年 月 日
Date of Application:

2001年12月14日

出 願 番 号
Application Number:

特願2001-381409

ST.10/C]:

[JP2001-381409]

出 願 人
Applicant(s):

セイコーエプソン株式会社

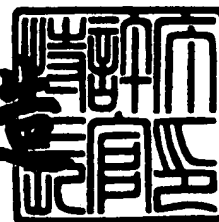
U.S. Appln. Filed 2-15-02
Inventor. M. Oyanagi
Mattingly Stanger & Malur
Docket NGB-103

CERTIFIED COPY OF
PRIORITY DOCUMENT

2002年 1月25日

特 許 庁 長 官
Commissioner,
Japan Patent Office

及 川 耕 造



出証番号 出証特2002-3000663

【書類名】 特許願

【整理番号】 13293901

【提出日】 平成13年12月14日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 3/00

【発明の名称】 マルチファンクションプリンタ

【請求項の数】 13

【発明者】

【住所又は居所】 長野県諏訪市大和三丁目3番5号 セイコーエプソン株式会社内

【氏名】 小 柳 誠

【特許出願人】

【識別番号】 000002369

【住所又は居所】 東京都新宿区西新宿二丁目4番1号

【氏名又は名称】 セイコーエプソン株式会社

【代理人】

【識別番号】 100075812

【弁理士】

【氏名又は名称】 吉 武 賢 次

【選任した代理人】

【識別番号】 100088889

【弁理士】

【氏名又は名称】 橘 谷 英 俊

【選任した代理人】

【識別番号】 100082991

【弁理士】

【氏名又は名称】 佐 藤 泰 和

【選任した代理人】

【識別番号】 100096921

【弁理士】

【氏名又は名称】 吉 元 弘

【選任した代理人】

【識別番号】 100103263

【弁理士】

【氏名又は名称】 川 崎 康

【選任した代理人】

【識別番号】 100107582

【弁理士】

【氏名又は名称】 関 根 毅

【先の出願に基づく優先権主張】

【出願番号】 特願2001- 43487

【出願日】 平成13年 2月20日

【手数料の表示】

【予納台帳番号】 087654

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 0103098

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 マルチファンクションプリンタ

【特許請求の範囲】

【請求項1】

スキャナ部とプリンタ部とを有するマルチファンクションプリンタであって、
前記スキャナ部が動作中であっても、前記プリンタ部のみをリセットする、リ
セット手段を、備えることを特徴とするマルチファンクションプリンタ。

【請求項2】

コンピュータから、前記プリンタ部をリセットするプリンタ部リセットコマン
ドを受信する、受信手段をさらに備えるとともに、
前記プリンタ部リセットコマンドを受信した場合に、前記リセット手段は、前
記プリンタ部のみのリセットを行う、
ことを特徴とする請求項1に記載のマルチファンクションプリンタ。

【請求項3】

前記マルチファンクションプリンタは、中央処理装置を1つだけ備えているこ
とを特徴とする請求項1又は請求項2に記載のマルチファンクションプリンタ。

【請求項4】

前記リセット手段は、少なくとも、
前記プリンタ部リセットコマンドを受信した場合には、コンピュータから送信
された印刷データを格納する入力バッファに、印刷データとしては無意味なデー
タと解釈されるダミーデータを印刷データとして書き込む、第1タスクと、
前記入力バッファに格納されている前記印刷データを読み出して、イメージデ
ータに展開する、第2タスクとを、
前記中央処理装置で実行することにより実現される、ことを特徴とする請求項
3に記載のマルチファンクションプリンタ。

【請求項5】

前記リセット手段は、前記中央処理装置で実行されるタスクとして、
前記プリンタ部リセットコマンドを受信した場合に、前記プリンタ部がリセッ
ト処理中であることを示すリセット中識別情報をセットする、第3タスクをさら

に備えることを特徴とする請求項4に記載のマルチファンクションプリンタ。

【請求項6】

前記第2タスクは、最も上位のメインループ処理と、このメインループ処理から階層的に派生する下位のループ処理とを有しており、前記下位のループ処理は、前記リセット中識別情報がセットされている場合には、その処理を抜けて上位のループ処理に戻るよう構成されている、ことを特徴とする請求項5に記載のマルチファンクションプリンタ。

【請求項7】

前記第2タスクでは、その時点のモードがキャラクタデータを印刷するキャラクタモードである場合には、前記タミーデータが所定回数以上連続したかどうかを判断する連続回数判断処理を行い、その時点のモードがグラフィックデータを印刷するグラフィックモードである場合には、この連続回数判断処理を行わない、ことを特徴とする請求項6に記載のマルチファンクションプリンタ。

【請求項8】

受信した印刷データを展開するデータ展開処理手段であって、前記リセット手段が前記プリンタ部リセットコマンドを受信した場合には、印刷データの展開処理を途中終了する、データ展開処理手段と、

前記データ展開処理手段により起動される、下位の処理手段であって、起動された後に所定時間経過した場合は、その処理が終了していない場合であっても、前記データ展開処理手段に戻る、下位の処理手段と、

をさらに備えることを特徴とする請求項2に記載のマルチファンクションプリンタ。

【請求項9】

前記データ展開処理手段の印刷データの展開処理が途中終了した場合に、プリントエンジンを初期状態に戻すプリントエンジン初期化手段を、さらに備えることを特徴とする請求項8に記載のマルチファンクションプリンタ。

【請求項10】

前記プリントエンジン初期化手段がプリントエンジンを初期状態に戻した後に、前記データ展開手段の設定状態を初期状態に戻す、ことを特徴とする請求項9

に記載のマルチファンクションプリンタ。

【請求項 11】

スキャナ部とプリンタ部とを有するマルチファンクションプリンタの制御方法であって、

前記スキャナ部が動作中であっても、前記プリンタ部のみをリセットする、ことを特徴とするマルチファンクションプリンタの制御方法。

【請求項 12】

スキャナ部とプリンタ部とを有するマルチファンクションプリンタに、

前記スキャナ部が動作中であっても、前記プリンタ部のみをリセットする、リセットステップを、

を実行させるためのプログラム。

【請求項 13】

スキャナ部とプリンタ部とを有するマルチファンクションプリンタに、

前記スキャナ部が動作中であっても、前記プリンタ部のみをリセットする、リセットステップを、

を実行させるためのプログラムを記録した記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、スキャナ部とプリンタ部とを有するマルチファンクションプリンタに関し、特に、スキャナ部とプリンタ部とが独立に動作可能なマルチファンクションプリンタに関する。

【0002】

【従来の技術】

スキャナ部とプリンタ部とが一体化されて、1つの筐体に格納されたマルチファンクションプリンタが普及してきている。このようなマルチファンクションプリンタにおいては、1台で、スキャナとしての役割と、プリンタとしての役割と、コピー機としての役割とを、果たすことができる。この場合、カラー印刷やカラーコピーを可能にするため、プリンタ部としては、いわゆるカラーインクジェ

ットプリンタが用いられている場合が多い。

【0003】

このようなマルチファンクションプリンタにおいて、ユーザが何らかの理由により、プリンタ部で印刷中のジョブをキャンセルしたい場合がある。この場合、ユーザは、マルチファンクションプリンタに接続されたコンピュータから、リセットコマンドを送信し、マルチファンクションプリンタでは、このリセットコマンドに基づいて、印刷中のジョブのキャンセルを行う。

【0004】

【発明が解決しようとする課題】

しかしながら、従来のマルチファンクションプリンタにおいては、このキャンセルにあたって、印刷を実行するために動作している各種のタスクを初期状態に戻すために、リブートの処理を実行していた。すなわち、マルチファンクションプリンタ上の各種タスクは、リアルタイムOS (Operating System) 上で動作しているため、このリアルタイムOSそのものをリブートすることにより、マルチファンクションプリンタを初期化して、印刷中のジョブをキャンセルしていた。

【0005】

しかし、スキャナ部が、プリンタ部と独立して動作可能なマルチファンクションプリンタもある。このようなマルチファンクションプリンタにおいては、プリンタ部がコンピュータから受信した印刷データの印刷をしつつ、これと並行して、スキャナ部が原稿等のスキャン動作を行うことが可能になる。このように並行動作している最中に、リアルタイムOSそのものをリブートしてしまうと、本来はユーザがリセットするつもりのないスキャナ部までもが、リセットされてしまう。

【0006】

しかも、ユーザが再度スキャンをやり直そうとして、コマンドをコンピュータからマルチファンクションプリンタに送信すると、リアルタイムOSがリブートして立ち上がり直されているので、通信エラーを起こしてしまう場合も生じてしまう。この場合、ユーザは、再度、コンピュータのスキャナ用のドライバを立ち上げ直して、通信エラーを解消しなければならない。

【 0 0 0 7 】

そこで本発明は、前記課題に鑑みてなされたものであり、プリンタ部とスキャナ部とを有するマルチファンクションプリンタにおいて、プリンタ部で行われている印刷動作のみを、キャンセルすることができるようにすることを目的とする。

【 0 0 0 8 】

【課題を解決するための手段】

上記課題を解決するため、本発明に係るマルチファンクションプリンタは、スキャナ部とプリンタ部とを有するマルチファンクションプリンタであって、前記スキャナ部が動作中であっても、前記プリンタ部のみをリセットする、リセット手段を、備えることを特徴とする。これにより、スキャナ部とプリンタ部とが並行して動作している場合にも、プリンタ部のみをリセットすることができるようになる。

【 0 0 0 9 】

この場合、コンピュータから、前記プリンタ部をリセットするプリンタ部リセットコマンドを受信する、受信手段をさらに備えるとともに、前記プリンタ部リセットコマンドを受信した場合に、前記リセット手段は、前記プリンタ部のみのリセットを行うようにしてもよい。これにより、ユーザは、コンピュータからプリンタ部リセットコマンドを送信することにより、リセットすることができるようになる。

【 0 0 1 0 】

さらにこの場合、前記マルチファンクションプリンタは、中央処理装置を1つだけ備えようとしてもよい。

【 0 0 1 1 】

さらに、前記リセット手段は、少なくとも、前記プリンタ部リセットコマンドを受信した場合には、コンピュータから送信された印刷データを格納する入力バッファに、印刷データとしては無意味なデータと解釈されるダミーデータを印刷データとして書き込む、第1タスクと、前記入力バッファに格納されている前記印刷データを読み出して、イメージデータに展開する、第2タスクとを、前記中

央処理装置で実行することにより実現されるようにしてもよい。このように、入力バッファにダミーデータを格納しておくことにより、入力バッファから印刷データを読み出してイメージデータに展開する第2タスクは、どのような処理をしていたとしても、いずれ、その展開処理が終わることになる。

【0012】

さらに、前記リセット手段は、前記中央処理装置で実行されるタスクとして、前記プリンタ部リセットコマンドを受信した場合に、前記プリンタ部がリセット処理中であることを示すリセット中識別情報をセットする、第3タスクを備えるようにしてもよい。これにより、このリセット中識別情報をチェックすることにより、各タスクがその時点でプリンタ部がリセット処理中であることを識別できるようになる。

【0013】

また、前記第2タスクは、最も上位のメインループ処理と、このメインループ処理から階層的に派生する下位のループ処理とを有しており、前記下位のループ処理は、前記リセット中識別情報がセットされている場合には、その処理を抜けて上位のループ処理に戻るよう構成してもよい。これにより、プリンタ部がリセット処理中である場合には、第2タスクに含まれている下位のループ処理は、順次、その処理を抜けて上位のループ処理に移行し、やがて、最も上位のメインループ処理に戻ることができる。

【0014】

さらに、前記第2タスクでは、その時点のモードがキャラクタデータを印刷するキャラクタモードである場合には、前記ダミーデータが所定回数以上連続したかどうかを判断する連続回数判断処理を行い、その時点のモードがグラフィックデータを印刷するグラフィックモードである場合には、この連続回数判断処理を行わないようにしてもよい。これにより、ダミーデータと同じ印刷データが出現しやすいキャラクタモードにおけるスループットの低下を抑止することができる。

【0015】

一方、本発明に係るマルチファンクションプリンタは、受信した印刷データを

展開するデータ展開処理手段であって、前記リセット手段が前記プリンタ部リセットコマンドを受信した場合には、印刷データの展開処理を途中終了する、データ展開処理手段と、前記データ展開処理手段により起動される、下位の処理手段であって、起動された後に所定時間経過した場合は、その処理が終了していない場合であっても、前記データ展開処理手段に戻る、下位の処理手段と、をさらに備えるようにしてもよい。このようにすれば、上述したようなダミーデータの書き込みを行わずとも、プリンタ部のみをリセットすることができるようになる。

【 0 0 1 6 】

さらに、本発明に係るマルチファンクションプリンタは、前記データ展開処理手段の印刷データの展開処理が途中終了した場合に、プリントエンジンを初期状態に戻すプリントエンジン初期化手段を、備えるようにしてもよい。またこの場合、前記プリントエンジン初期化手段がプリントエンジンを初期状態に戻した後に、前記データ展開手段の設定状態を初期状態に戻すようにしてもよい。

【 0 0 1 7 】

なお、本発明は上述したようなマルチファンクションプリンタを実現するための制御方法としてとらえることもでき、また、そのためのプログラム及びそのプログラムが記録された記録媒体としてとらえることもできる。

【 0 0 1 8 】

【発明の実施の形態】

〔第 1 実施形態〕

本発明は、プリンタ部だけをリセットするプリンタ部リセットコマンドを受信したことを、必要なタスクに通知するとともに、印刷処理に関する各種タスクがその時点で行っている処理から抜け出せるように、印刷データを格納する入力バッファに、印刷データの解釈上は意味のないダミーデータを連続して書き込むようにしたものである。そして、これにより、印刷処理に関する各種の複雑なタスクが、その時点で行っている処理から自ら抜け出して、プリンタ部のみをリセットすることができるようにしたものである。以下、本発明に係る一実施形態を詳細に説明する。

【 0 0 1 9 】

まず、図 1 に基づいて、本実施形態に係るマルチファンクションプリンタ 1 0 の内部構成を説明する。この図 1 は、スキャナ部とプリンタ部とが一体化されたマルチファンクションプリンタ 1 0 の内部構成を示すブロック図である。

【 0 0 2 0 】

図 1 に示すように、マルチファンクションプリンタ 1 0 は、RAM (Random Access Memory) 1 2 と、ROM (Read Only Memory) 1 4 と、CPU (Central Processing Unit: 中央処理装置) 1 6 と、スキャナスキャナASIC (Application Specific IC) 1 8 と、プリンタASIC 2 0 とを備えており、これらは内部バスを介して、相互に接続されている。このことから分かるように、本実施形態に係るマルチファンクションプリンタ 1 0 においては、CPU は 1 つだけ設けられている。

【 0 0 2 1 】

スキャナASIC 1 8 には、実際のスキャン動作を行うスキャナ機構部 2 2 が接続されている。また、プリンタASIC 2 0 には、実際に印刷動作を行うプリントエンジン 2 4 が接続されている。

【 0 0 2 2 】

さらに、マルチファンクションプリンタ 1 0 は、USB ケーブルを介してコンピュータに接続される USB (Universal Serial Bus) ポート 2 6 を備えている。この USB ポート 2 6 は USB ハブ 2 8 に接続されており、USB ハブ 2 8 はスキャナASIC 1 8 とプリンタASIC 2 0 とに接続されている。したがって、コンピュータから送信されたデータやコマンドは、USB ハブ 2 8 を介して、スキャナASIC 1 8 とプリンタASIC 2 0 の双方に入力される。

【 0 0 2 3 】

例えば、コンピュータからマルチファンクションプリンタ 1 0 に送信された印刷データは、プリンタASIC 2 0 で受信され、RAM 1 2 に形成された入力バッファに格納される。入力バッファに格納された印刷データは、順次、イメージデータに展開されて、プリンタASIC 2 0 を介して、プリントエンジン 2 4 で印刷される。本実施形態に係るプリントエンジン 2 4 は、カラーのインクジェットヘッドを有しており、このヘッドをキャリッジに搭載して主走査方向に交互に

移動することにより、カラー印刷が可能になっている。

【0024】

一方、スキャナ機構部10は、光学的に原稿を読み取るラインイメージセンサを有している。このラインイメージセンサは、キャリッジに搭載されており、キャリッジを原稿の一端側から他端側まで移動させることにより、原稿全体を読み取ることが可能である。この読み取り動作は、スキャナASIC12が制御しており、読み取ったスキャンデータは、RAM12に形成されたスキャナバッファに格納される。

【0025】

本実施形態に係るマルチファンクションプリンタ10においては、スキャナ部を構成するスキャナ機構部22におけるスキャン動作と、プリンタ部を構成するプリントエンジン24における印刷動作とが、並行して行えるようになっている。例えば、スキャナ機構部22で原稿のスキャン動作をしながら、プリントエンジン24でコンピュータから送信されてきている印刷データの印刷を、順次、行うことが可能である。また、本実施形態に係るマルチファンクションプリンタ10においては、スキャナ部とプリンタ部とが並行して動作している状態で、プリンタ部のみをリセット（印刷ジョブのキャンセル）することが可能である。

【0026】

このプリンタ部のみをリセットする際の全体的な処理の流れを、図2に基づいて説明する。この図2は、プリンタ部をリセットし、スキャナ部の動作は継続させるためのプリンタ部リセットコマンドを、コンピュータから受信した場合のマルチファンクションプリンタ10の処理を説明するタイミングチャートである。

【0027】

図2に示すように、本実施形態に係るマルチファンクションプリンタ10は、インターフェース制御タスクT10と、制御コマンド解釈タスクT20と、メインタスクT30と、ジョブコントロールタスクT40とを、備えている。なお、これ以外のタスクも、マルチファンクションプリンタ10は備えているが、ここでは、本実施形態に関係のあるタスクのみを示している。また、これらのタスクは、このマルチファンクションプリンタ10の備えるリアルタイムOS上で管理

されており、CPU16で実行される。

【0028】

まず、ユーザが印刷中のジョブをキャンセルするために、コンピュータから、プリンタ部リセットコマンドを、マルチファンクションプリンタ10に送信する。このプリンタ部リセットコマンドは、インターフェース制御タスクT10により受信される。インターフェース制御タスクT10は、受信したコマンドが制御系のコマンドであるか、印刷データ等であるのかを判断し、この例では制御系のコマンドであるので、このプリンタ部リセットコマンドの解釈を要求すべく、これを制御コマンド解釈タスクT20に送信する。

【0029】

これを受信した制御コマンド解釈タスクT20が、これを解釈することにより、受信したコマンドがプリンタ部リセットコマンドであることが判明する。このため、制御コマンド解釈タスクT20は、インターフェース制御タスクT10にリプライを依頼し、インターフェース制御タスクT10はこれに基づいてリプライをコンピュータに返信する。また、制御コマンド解釈タスクT20は、リセットコマンド受信メッセージを、ジョブコントロールタスクT40に送信する。

【0030】

リセットコマンド受信メッセージを受け取ったジョブコントロールタスクT40は、リセット中フラグRSFをセットするとともに、リセット処理移行要求を、インターフェース制御タスクT10へ送信する。そして、ジョブコントロールタスクT40は、それ以降の印刷、紙送り要求などのジョブを、徹底的に読み捨てる。但し、応答が必要なメッセージについては、とりあえず必要な返信のみ行っておく。

【0031】

このリセット処理移行要求を受け取ったインターフェース制御タスクT10は、印刷データを格納しておく入力バッファの全領域に16進数の「00」である0x00をダミーデータとして書き込み、受信データ管理タスクに、データが到着した旨を通知する。

【0032】

プリンタ部が印刷中の場合、メインタスクT30では、様々なループ処理を行っているが、印刷データの先頭にはデータ量を示す所定のコマンドが付加されており、受信データ管理タスクはコマンドに付加されたデータ量を受信するまでは待機しつづける。この待機状態を終了させるためには、コマンドに付加されたデータ量を少なくとも受信する必要がある。そこで、本実施形態においては、コマンドに付加されたデータ量を少なくとも満たすデータ量の0x00、つまり、コマンドとして通常存在しない無意味なNULLデータを書き込むことにより、この待機状態を終了させる。これにより、他のコマンドと混同することなく、リセット処理が可能になる。

【0033】

メインタスクで取得すべき印刷データが、入力バッファになくなった場合は、受信データ管理タスク（図示省略）を経由して、印刷データ獲得要求がインターフェース制御タスクT10に発行される。この場合も、インターフェース制御タスクT10は、0x00のダミーの印刷データを入力バッファに書き込み、受信データ管理タスクに、データが到着した旨を通知する。

【0034】

このように、メインタスクT30は、NULLデータのみを取得することになるので、いずれ下位のループ処理を抜けて、メインタスクT30の最上位のループ処理に戻ってくることができる。最上位のループ処理に戻った場合には、メインタスクT30は、ジョブコントロールタスクT40に対して、リセット処理終了確認通知を送信し、メインタスクT30が最上位のループ処理に戻ったことを通知する。

【0035】

リセット処理終了確認通知を受け取ったジョブコントロールタスクT40は、リセット中フラグRSFをリセットし、印刷や紙送り要求などのジョブの読み捨てを中止する。また、リセット処理解除要求を、インターフェース制御タスクT10へ送信する。これにより、インターフェース制御タスクT10は、リセット処理モードから、通常の印刷データを受信するモードに戻る。一方、ジョブコントロールタスクT40は、メインタスクT30へ、リセット処理終了確認通知に

対する返信を行う。この返信がメインタスク T 3 0 に戻ってきた場合、メインタスク T 3 0 は、必要な設定状態を初期状態に戻す。これにより、プリンタ部のみのリセットが行われたことになる。

【 0 0 3 6 】

以上、マルチファンクションプリンタ 1 0 がプリンタ部リセットコマンドを受信した場合の処理を概略的に説明したが、次に、上述した各タスク毎にその詳細な処理内容を説明する。

【 0 0 3 7 】

図 3 は、上述したインターフェース制御タスク T 1 0 で行われるインターフェース制御処理を説明するフローチャートである。本実施形態においては、このインターフェース制御処理は、ROM 1 4 に格納されているインターフェース制御プログラムを、リアルタイム OS 上で CPU 1 6 が実行することにより実現される。

【 0 0 3 8 】

図 3 に示すように、インターフェース制御タスク T 1 0 は、まず、何からのメッセージを受信したかどうかを判断する（ステップ S 1 0）。何らメッセージを受信していない場合（ステップ S 1 0 : N o）には、このステップ S 1 0 の処理を繰り返して待機する。

【 0 0 3 9 】

何らかのメッセージを受信した場合（ステップ S 1 0 : Y e s）には、そのメッセージが、ジョブコントロールタスク T 4 0 から送信されたりセット処理移行要求であるかどうかを判断する（ステップ S 1 1）。受信したメッセージがセット処理移行要求である場合（ステップ S 1 1 : Y e s）には、インターフェース制御タスク T 1 0 は、入力バッファの全領域に 0 x 0 0 を書き込む（ステップ S 1 2）。

【 0 0 4 0 】

次に、インターフェース制御タスク T 1 0 は、メインタスク T 3 0 に、入力バッファに受信データを格納した旨を通知する（ステップ S 1 3）。上述したように、この通知は、受信データ管理タスクを介して、メインタスク T 3 0 に送信さ

れる。

【0041】

次に、インターフェース制御タスクT10は、リセット処理中フラグWSFをセットする（ステップS14）。このリセット処理中フラグWSFは、インターフェース制御タスクT10が、現在、プリンタ部がリセット処理モードにあるか、通常の印刷データを受信するモードにあるのかを、区別するためのフラグである。そして、インターフェース制御タスクT10は、上述したステップS10に戻る。

【0042】

一方、上述したステップS11で、受信したメッセージがリセット処理移行要求でないと判断した場合（ステップS11：No）には、受信したメッセージが、メインタスクT30からの印刷データの要求であるかどうかを判断する（ステップS15）。上述したように、このメッセージは、メインタスクT30から、受信データ管理タスクを介して、受信される。

【0043】

受信したメッセージが印刷データの要求である場合（ステップS15：Yes）には、インターフェース制御タスクT10は、リセット処理中フラグWSFをチェックして、現在、リセット処理中であるかどうかを判断する（ステップS16）。リセット処理中でない場合（ステップS16：No）には、メインタスクT30が印刷データの受信待ちである旨をセットして（ステップS17）、上述したステップS10に戻る。

【0044】

一方、ステップS16で、現在、リセット処理中であると判断した場合（ステップS16：Yes）には、入力バッファに0x00を書き込むべく、ステップS12からの処理を実行する。

【0045】

これに対して、上述したステップS15で、受信したメッセージが、メインタスクT30からの印刷データの要求でないと判断した場合（ステップS15：No）には、受信したメッセージが、ジョブコントロールタスクT40からのリセ

ット処理解除要求であるかどうかを判断する（ステップS18）。受信したメッセージが、リセット処理解除要求である場合（ステップS18：Yes）には、インターフェース制御タスクT10は、入力バッファへの0x00書き込みを停止する（ステップS19）。続いて、リセット処理中フラグWSFをリセットして（ステップS20）、上述したステップS10に戻る。

【0046】

一方、上述したステップS18で、受信したメッセージがリセット処理解除要求でないと判断した場合（ステップS18：No）には、該当するメッセージについて必要な処理を実行する（ステップS21）。例えば、上述したように、コンピュータからのプリンタ部リセットコマンドなどの制御系のコマンドを受信した場合には、これを制御コマンド解釈タスクT20へ転送する。また、例えば、制御コマンド解釈タスクT20からプリンタ部リセットコマンドに対するリプライの要求を受信した場合には、そのリプライをコンピュータへ転送する。そして、上述したステップS10に戻る。

【0047】

次に、図4に基づいて、制御コマンド解釈タスクT20について説明する。図4は、上述した制御コマンド解釈タスクT20で行われる制御コマンド解釈処理を説明するフローチャートである。本実施形態においては、この制御コマンド解釈処理は、ROM14に格納されている制御コマンド解釈プログラムを、リアルタイムOS上でCPU16が実行することにより実現される。

【0048】

図4に示すように、制御コマンド解釈タスクT20は、まず、何からのメッセージを受信したかどうかを判断する（ステップS30）。何らメッセージを受信していない場合（ステップS30：No）には、このステップS30の処理を繰り返して待機する。

【0049】

何らかのメッセージを受信した場合（ステップS30：Yes）には、そのメッセージが、インターフェース制御タスクT10を介して、コンピュータから送信されたプリンタ部リセットコマンドであるかどうかを判断する（ステップS3

1)。受信したメッセージがプリンタ部リセットコマンドである場合（ステップ S 3 1 : Y e s）には、コンピュータに対する応答データ列を作成し（ステップ S 3 2）、これをインターフェース制御タスク T 1 0 に送信して、リプライを依頼する（ステップ S 3 3）。

【 0 0 5 0 】

次に、制御コマンド解釈タスク T 2 0 は、ジョブコントロールタスク T 4 0 へ、リセットコマンド受信をメッセージとして送信する（ステップ S 3 4）。そして、上述したステップ S 3 0 に戻る。

【 0 0 5 1 】

一方、上述したステップ S 3 1 で、受信したメッセージがプリンタ部リセットコマンドでないと判断した場合（ステップ S 3 1 : N o）には、該当するメッセージについて必要な解釈を行い、必要な処理を実行する（ステップ S 3 5）。例えば、紙送り要求のメッセージである場合は、その解釈を行い、必要なメッセージをジョブコントロールタスク T 4 0 等へ送信する。そして、上述したステップ S 3 0 の処理に戻る。

【 0 0 5 2 】

次に、図 5 及び図 6 に基づいて、メインタスク T 3 0 について説明する。図 5 は、上述したメインタスク T 3 0 で行われるデータ展開制御処理を説明するフローチャートであり、本実施形態におけるメインループ処理を示している。図 6 は、このデータ展開制御処理から派生する下位のループ処理であるグラフィックマネージャ処理を説明するフローチャートである。本実施形態におけるメインタスク T 3 0 には、グラフィックマネージャ処理以外にも、数多くの下位のループ処理が存在するが、ここでは、グラフィックマネージャ処理を代表例として説明する。また、下位のループ処理から派生するさらに下位のループ処理も存在する。つまり、本実施形態においては、メインタスク T 3 0 のループ処理は階層的に構成されている。

【 0 0 5 3 】

なお、図 5 のデータ展開制御処理はキャラクタ系の印刷データを処理するモードであり、図 6 のグラフィックマネージャ処理はグラフィック系の印刷データを

処理するモードである。また、本実施形態においては、これらのデータ展開制御処理及びグラフィックマネージャ処理は、ROM14に格納されているデータ展開制御プログラム及びグラフィックマネージャプログラムを、リアルタイムOS上でCPU16が実行することにより実現される。

【0054】

図5に示すように、メインタスクT30は、まず、入力バッファにデータを受信したかどうかを判断する（ステップS40）。データを受信していない場合（ステップS40：No）には、このステップS40の処理を繰り返して待機する。

【0055】

入力バッファにデータを受信した場合（ステップS40：Yes）には、メインタスクT30は、入力バッファからデータを1バイト読み込む（ステップS41）。次に、この読み込んだデータが、印刷モード移行要求であるかどうかを判断する（ステップS42）。本実施形態に係るマルチファンクションプリンタ10は、モードとして、キャラクタ系の印刷データを印刷するキャラクタモードと、グラフィック系の印刷データを印刷するグラフィックモードとを備えており、両者の間は印刷モード移行要求を受信することにより、切り替わるようになっている。

【0056】

データが印刷モード移行要求でない場合（ステップS42：No）には、取得したデータが0x00であるかどうかを判断する（ステップS43）。取得したデータが0x00でない場合（ステップS43：No）には、通常の印刷であると考えられるので、キャラクターモードの通常の処理を実行する（ステップS44）。そして、上述したステップS40の処理に戻る。

【0057】

一方、取得したデータが0x00である場合（ステップS43：Yes）には、この0x00が所定回数以上（例えば、1024回＝1KB）、連続したかどうかを判断する（ステップS45）。ここで、所定回数以上、0x00が連続したかどうかのチェックをするのは、キャラクターモードで1バイト毎にリセット中

フラグ R S F の確認をすると、通常のキャラクタ印刷の際のスループットが低下してしまうので、これを回避するためである。換言すれば、キャラクタ印刷の場合、印刷データとして 0 x 0 0 を受信することも少なからずあり得るからである。但し、多少のスループットの低下を許容して、このような判断をしないようにしてもよい。

【 0 0 5 8 】

したがって、0 x 0 0 が所定回数以上連続していない場合（ステップ S 4 5 : N o）には、上述した通常のキャラクタモードの処理を行う（ステップ S 4 4）。一方、0 x 0 0 が所定回数以上連続した場合（ステップ S 4 5 : Y e s）には、リセット中フラグ R S F がセットされているかどうかを判断する（ステップ S 4 6）。このリセット中フラグ R S F は、ジョブコントロールタスク T 4 0 が管理するフラグであるので、このステップ S 4 6 の処理は、同一タスク内のフラグを確認する処理と比べて、処理時間が長くなる。

【 0 0 5 9 】

リセット中フラグ R S F がセットされていない場合（ステップ S 4 6 : N o）には、上述したステップ S 4 0 の処理に戻る。一方、リセット中フラグ R S F がセットされている場合（ステップ S 4 6 : Y e s）には、ジョブコントロールタスク T 4 0 に、リセット処理終了確認通知を送信する（ステップ S 4 7）。なぜなら、本実施形態においては、このデータ展開制御処理が、最も上位のループ処理だからである。

【 0 0 6 0 】

次に、メインタスク T 3 0 は、ジョブコントロールタスク T 4 0 からのリセット処理終了確認通知に対する返信を待つ（ステップ S 4 8）。返信があった場合には、このメインタスク T 3 0 内の各種の設定状態を、初期状態に設定する（ステップ S 4 9）。これにより、メインタスク T 3 0 のリセットが完了し、上述したステップ S 4 0 の処理に戻ることになる。

【 0 0 6 1 】

次に、図 6 に示すグラフィックマネージャ処理について説明する。このグラフィックマネージャ処理は、図 5 に示すステップ S 5 0 のグラフィックモードに移

行した場合のループである。したがって、このグラフィックマネージャ処理が終了すると、図5のステップS51の処理に戻ることになる。

【0062】

図6に示すように、メインタスクT30のグラフィックマネージャ処理では、まず、入力バッファに受信データがあるかどうかを判断する（ステップS60）。受信データがない場合（ステップS60：No）には、このステップS60の処理を繰り返して待機する。

【0063】

受信データがある場合（ステップS60：Yes）には、入力バッファから1バイトのデータを読み込む（ステップS61）。次に、メインタスクT30は、この取得したデータが、0x00であるかどうかを判断する（ステップS62）。取得したデータが、0x00でない場合（ステップS62：No）には、通常のグラフィックモードの処理を行う（ステップS63）。そして、上述したステップS60に戻る。

【0064】

一方、ステップS62で、取得したデータが0x00である場合（ステップS62：Yes）には、ジョブコントロールタスクT40の管理するリセット中フラグRSFが、セットされているかどうかを判断する（ステップS64）。リセット中フラグRSFがセットされていない場合（ステップS64：No）には、上述したステップS60に戻る。

【0065】

一方、リセット中フラグRSFがセットされている場合（ステップS64：Yes）には、グラフィックマネージャ処理から、図5のデータ展開制御処理に戻る（ステップS65）。

【0066】

なお、このグラフィックマネージャ処理では、0x00が所定回数以上連続したかどうかの判断はしていない。これは、グラフィックモードで印刷データとして0x00を受信することは、極めて希であり、0x00を検出するごとにリセット中フラグRSFを確認しに行っても、特にスループットが低下するとは考え

られないからである。但し、このような判断をするようにしてもよい。

【 0 0 6 7 】

また、本実施形態においては、データ展開制御処理（メインループ処理）の下位のループ処理は、グラフィックマネージャ処理以下にも、数多く存在するが、これらの下位のループ処理にも、ステップ S 6 2 及びステップ S 6 4 に相当する処理が、埋め込まれている。また、ループ処理のレベルがさらに深い階層構造になっている場合もある。この場合でも、各ループ処理にステップ S 6 2 及びステップ S 6 4 に相当する処理が埋め込まれている。但し、この場合、ステップ S 6 5 で戻るべきループ処理は、1 つだけ上位のループ処理である。したがって、取得したデータが 0 x 0 0 であり、且つ、リセット中フラグ R S F がセットされている場合には、各ループ処理は、リセット処理中であると認識して、順次、上位のループ処理に戻ってきて、最終的に、最も上位のループ処理である図 5 のデータ展開制御処理に辿り着くことになる。

【 0 0 6 8 】

次に、図 7 及び図 8 に基づいて、ジョブコントロールタスク T 4 0 について説明する。図 7 は、上述したジョブコントロールタスク T 4 0 で行われるジョブコントロール処理を説明するフローチャートであり、図 8 は、ジョブコントロール処理の下位のループ処理であるネゴシエーション処理を説明するフローチャートである。

【 0 0 6 9 】

本実施形態においては、これらのジョブコントロール処理及びネゴシエーション処理は、ROM 1 4 に格納されているジョブコントロール処理プログラム及びネゴシエーションプログラムを、リアルタイム OS 上で CPU 1 6 が実行することにより実現される。

【 0 0 7 0 】

図 7 に示すように、ジョブコントロールタスク T 4 0 は、まず、何からのメッセージを受信したかどうかを判断する（ステップ S 7 0）。何らメッセージを受信していない場合（ステップ S 7 0 : N o）には、このステップ S 7 0 の処理を繰り返して待機する。

【 0 0 7 1 】

何らかのメッセージを受信した場合（ステップ S 7 0 : Y e s）には、リセット中フラグ R S F がセットされているかどうかを判断する（ステップ S 7 1）。リセット中フラグ R S F がセットされていない場合（ステップ S 7 1 : N o）には、受信したメッセージが、制御コマンド解釈タスク T 2 0 から送信されたリセットコマンド受信のメッセージであるかどうかを判断する（ステップ S 7 2）。

【 0 0 7 2 】

受信したメッセージがリセットコマンド受信のメッセージでない場合（ステップ S 7 2 : N o）には、そのメッセージについて必要なジョブを実行し、そのコントロールを行う（ステップ S 7 3）。そして、上述したステップ S 7 0 に戻る。

【 0 0 7 3 】

一方、上述したステップ S 7 2 で、受信したメッセージがリセットコマンド受信のメッセージである場合（ステップ S 7 2 : Y e s）には、ジョブコントロールタスク T 4 0 は、リセット中フラグ R S F をセットする（ステップ S 7 4）。続いて、ジョブコントロールタスク T 4 0 は、インターフェース制御タスク T 1 0 へ、リセット処理移行要求を送信する（ステップ S 7 5）。そして、上述したステップ S 7 0 に戻る。

【 0 0 7 4 】

一方、上述したステップ S 7 1 の処理で、リセット中フラグ R S F がセットされていると判断した場合（ステップ S 7 1 : Y e s）には、受信したメッセージが、何らかの応答を必要としているメッセージであるかどうかを判断する（ステップ S 7 6）。応答が必要なメッセージでない場合（ステップ S 7 6 : N o）には、そのメッセージを読み捨てる（ステップ S 7 7）。メッセージを読み捨てることにより、ジョブが読み捨てられたことになる。そして、上述したステップ S 7 0 の処理に戻る。

【 0 0 7 5 】

一方、上述したステップ S 7 6 で、受信したメッセージが応答の必要なメッセージである場合（ステップ S 7 6 : Y e s）には、ジョブコントロールタスク T

40は、そのメッセージが、メインタスクT30からのリセット処理終了確認通知であるかどうかを判断する（ステップS78）。受信したメッセージが、リセット処理終了確認通知でない場合（ステップS78：No）には、メインタスクT30がまだ最上位のループ処理に戻ってきていないことになるので、とりあえず、受信したメッセージに対して必要な返事を送信しておく（ステップS79）。これは、メインタスクT30の下位のループ処理で、ジョブコントロールタスクT40からの返信待ちになっているものがある可能性があるため、このループ処理を終了させて上位のループ処理に戻すために、必要な返事のみは返信する必要があるからである。但し、ジョブコントロールタスクT40では、そのメッセージの内容は実行せずに、読み捨てる。そして、上述したステップS70の処理に戻る。

【0076】

一方、上述したステップS78で、受信したメッセージが、メインタスクT30からのリセット処理終了確認通知である場合（ステップS78：Yes）には、メインタスクT30が最上位のループに戻ってきたことになるので、リセット中フラグRSFをリセットする（ステップS80）。続いて、ネゴシエーション処理を行う（ステップS81）。なお、このネゴシエーション処理の詳細内容は後述する、

次に、ジョブコントロールタスクT40は、インターフェース制御タスクT10に、リセット処理解除要求を送信する（ステップS82）。続いて、ジョブコントロールタスクT40は、メインタスクT30に、リセット処理終了確認通知の返事を送信する（ステップS83）。そして、上述したステップS70に戻る。

【0077】

次に、図8に基づいて、上述した図7のステップS81のネゴシエーション処理について詳しく説明する。

【0078】

図8に示すように、まず、ネゴシエーション処理では、プリントエンジン24が動作中であるかどうかを判断する（ステップS90）。プリントエンジン24

が動作中である場合（ステップS90：Yes）には、このステップS90の処理を繰り返して、待機する。

【0079】

プリントエンジン24が動作中でない場合（ステップS90：No）には、ジョブコントロールタスクT40に残っているタスクが存在するかどうかを判断する（ステップS91）。ジョブコントロールタスクT40に残っているタスクがない場合（ステップS91：No）には、このネゴシエーション処理を終了する。

【0080】

一方、ジョブコントロールタスクT40に残っているタスクが存在する場合（ステップS91：Yes）には、そのジョブを捨てて（ステップS92）、メモリも解放する（ステップS93）。そして、このネゴシエーション処理を終了する。

【0081】

以上のように、本実施形態に係るマルチファンクションプリンタ10によれば、スキャナ部とプリンタ部とが並行して動作している場合でも、プリンタ部のみをリセットすることができるようにしたので、ユーザの利便性が向上する。すなわち、スキャナ部とプリンタ部とが並行して動作している場合に、プリンタ部をリセットしようとして、スキャナ部のスキャン動作までもが停止してしまう事態を回避することができる。

【0082】

具体的には、プリンタ部リセットコマンドを受信した場合には、インターフェース制御タスクT10が入力バッファにNULLデータを印刷データとして強制的に書き込むようにし、ジョブコントロールタスクT40がリセット処理中であることを示すリセット中フラグRSFをセットすることとした。このため、入力バッファから印刷データを読み出して展開処理を行っているメインタスクT30は、いずれその処理を終えることができる。すなわち、メインタスクT30が階層的なループ処理で構成されている場合、NULLデータに基づいて順次処理を行い、リセット中フラグRSFがセットされていることを検出すると、下位のル

ープ処理から順次抜け出して上位のループ処理に移行するので、最終的には、最上位のメインループ処理に自律的に戻ってくることができる。

【0083】

このため、複雑なループ処理で構成されている既存のメインタスクT30の処理内容を大きく変更することなく、本実施形態に係るマルチファンクションプリンタ10を実現することができる。

【0084】

〔第2実施形態〕

本発明の第2実施形態は、上述した第1実施形態を変形し、メインタスクT30により起動される下位のループ処理をタイマー設定して起動し、このタイマーに基づいて、下位のループ処理を起動してから一定時間経過した場合には、その下位のループ処理を途中終了して、メインタスクT30のメイン処理に戻るようにしたものである。そしてこれにより、インターフェース処理タスクT10がNULL書き込みを行わなくとも、マルチファンクションプリンタ10のプリンタ部のみをリセットすることができるようにしたものである。より詳しくを、以下に説明する。

【0085】

本実施形態に係るマルチファンクションプリンタ10の内部構成は、上述した図1と同様である。但し、コンピュータからプリンタ部リセットコマンドを受信した場合の処理が異なる。

【0086】

図9は、プリンタ部リセットコマンドを受信した場合のマルチファンクションプリンタ10の処理を説明するタイミングチャートである。図2と比べると、図9にはエンジン制御タスクT50が追加されている。但し、このエンジン制御タスクT50は、上述した第1実施形態のマルチファンクションプリンタ10でも備えているが、図2では省略しただけである。

【0087】

図9に示すように、コンピュータから送信されたプリンタ部リセットコマンドは、インターフェース制御タスクT10により受信される。インターフェース制

御タスク T 1 0 は、受信したコマンドが制御系のコマンドであるか、印刷データ等であるのかを判断し、この例では制御系のコマンドであるので、このプリンタ部リセットコマンドの解釈を要求すべく、これを制御コマンド解釈タスク T 2 0 に送信する。

【 0 0 8 8 】

これを受信した制御コマンド解釈タスク T 2 0 が、これを解釈することにより、受信したコマンドがプリンタ部リセットコマンドであることが判明する。このため、制御コマンド解釈タスク T 2 0 は、インターフェース制御タスク T 1 0 にリプライを依頼し、インターフェース制御タスク T 1 0 はこれに基づいてリプライをコンピュータに返信する。また、制御コマンド解釈タスク T 2 0 は、リセットコマンド受信メッセージを、ジョブコントロールタスク T 4 0 に送信する。

【 0 0 8 9 】

リセットコマンド受信メッセージを受け取ったジョブコントロールタスク T 4 0 は、リセット中フラグ R S F をセットするとともに、リセット処理実行要求を、メインタスク T 3 0 へ送信する。そして、ジョブコントロールタスク T 4 0 は、それ以降の印刷、紙送り要求などのジョブを、徹底的に読み捨てる。但し、応答が必要なメッセージについては、とりあえず必要な返信のみ行っておく。

【 0 0 9 0 】

このリセット処理実行要求を受け取ったメインタスク T 3 0 は、データ展開処理のために R A M 1 2 に確保していたメモリ領域を解放する。そして、メインタスク T 3 0 は、リセット処理終了通知をジョブコントロールタスク T 4 0 に送信する。なお、本実施形態においては、メインタスク T 3 0 から起動される下位のループ処理が実行されていた場合には、下位の処理はタイマーにより所定時間経過後にメインタスク T 3 0 のメイン処理に戻るよう設定されている。

【 0 0 9 1 】

リセット処理終了通知を受け取ったジョブコントロールタスク T 4 0 は、プリントエンジン 2 4 の機械的な状態を初期設定に戻すために、エンジン制御タスク T 5 0 に機械的状态確認メッセージを送信する。この機械的状态確認メッセージを受信したエンジン制御タスク T 5 0 は、プリントエンジン 2 4 の機械的な状態

を初期化して、初期状態に戻す。そして、エンジン制御タスクT50は、ジョブコントロールタスクT40に、機械的状态初期化終了メッセージを送信する。

【0092】

この機械的状态確認メッセージを受信したジョブコントロールタスクT40は、リセット中フラグRSFをリセットし、リセット処理終了通知の返信を、メインタスクT30に送信する。これを受けて、メインタスクT30では、このメインタスクT30内における必要な設定状態を初期状態に戻す。これにより、プリンタ部のみのリセットが行われたことになる。

【0093】

また、リセット中フラグRSFがセットされている間（つまり、図9の例では、リセット中フラグRSFが「1」の間）は、インターフェース制御タスクT10は、コンピュータから印刷データを受信した場合でも、その印刷データを送信したコンピュータに対して、データ受信不可のメッセージを返信する。つまり、リセット処理中には、印刷データを受け付けないようにする。

【0094】

以上、本実施形態に係るマルチファンクションプリンタ10がプリンタ部リセットコマンドを受信した場合の処理を概略的に説明したが、次に、上述した各タスク毎にその詳細な処理内容を説明する。

【0095】

図10は、上述したインターフェース制御タスクT10で行われるインターフェース制御処理を説明するフローチャートである。本実施形態においても、このインターフェース制御処理は、ROM14に格納されているインターフェース制御プログラムを、リアルタイムOS上でCPU16が実行することにより実現される。

【0096】

図10に示すように、インターフェース制御タスクT10は、まず、何からのメッセージを受信したかどうかを判断する（ステップS110）。何らメッセージを受信していない場合（ステップS110：No）には、このステップS110の処理を繰り返して待機する。

【0097】

何らかのメッセージを受信した場合（ステップS110：Yes）には、現在、リセット処理中であるかどうかを判断する（ステップS111）。具体的には、リセット中フラグRSFがセットされているかどうかに基づいて、現在リセット処理中であるかどうかを判断する。リセット処理中である場合（ステップS111：Yes）には、そのメッセージを処理することはできないので、そのメッセージを送信したコンピュータに、データ受信不可のメッセージを送信する（ステップS112）。そして、上述したステップS110に戻る。

【0098】

一方、ステップS111において、リセット処理中でないと判断した場合（ステップS111：No）には、該当するメッセージについて必要な処理を実行する（ステップS113）。例えば、上述したように、コンピュータからのプリンタ部リセットコマンドなどの制御系のコマンドを受信した場合には、これを制御コマンド解釈タスクT20へ転送する。また、例えば、制御コマンド解釈タスクT20からプリンタ部リセットコマンドに対するリプライの要求を受信した場合には、そのリプライをコンピュータへ転送する。そして、上述したステップS110に戻る。

【0099】

次に、制御コマンド解釈タスクT20についてであるが、本実施形態に係る制御コマンド解釈タスクT20の処理は、上述した第1実施形態における図4と同様の処理であるので、ここでは説明を省略する。

【0100】

次に、図11及び図12に基づいて、メインタスクT30について説明する。図11は、上述したメインタスクT30で行われるメインタスク処理を説明するフローチャートであり、本実施形態におけるメインループ処理を示している。図12は、このメインタスク処理から派生する下位のループ処理であるデータ展開処理を説明するフローチャートである。本実施形態におけるメインタスクT30には、このデータ展開処理以外にも、数多くの下位のループ処理が存在するが、ここでは、データ展開処理を代表例として説明する。また、下位のループ処理が

ら派生するさらに下位のループ処理も存在する。つまり、本実施形態においては、メインタスク T 3 0 のループ処理は階層的に構成されている。

【0101】

なお、本実施形態においても、これらのメインタスク処理及びデータ展開処理は、ROM 1 4 に格納されているメインタスクプログラム及びデータ展開プログラムを、リアルタイム OS 上で CPU 1 6 が実行することにより実現される。

【0102】

図 1 1 に示すように、メインタスク T 3 0 は、まず、何らかのメッセージを受信したかどうかを判断する（ステップ S 1 2 0）。何らメッセージを受信していない場合（ステップ S 1 2 0 : N o）には、このステップ S 1 2 0 の処理を繰り返して待機する。

【0103】

一方、何らかのメッセージを受信した場合（ステップ S 1 2 0 : Y e s）には、メインタスク T 3 0 は、そのメッセージが、ジョブコントロールタスク T 4 0 から送信されたリセット処理実行要求であるかどうかを判断する（ステップ S 1 2 1）。受信したメッセージがリセット処理実行要求でない場合（ステップ S 1 2 1 : N o）には、そのメッセージが印刷データに関するデータ展開処理要求であるかどうかを判断する（ステップ S 1 2 2）。

【0104】

受信したメッセージがデータ展開処理要求である場合（ステップ S 1 2 2 : Y e s）には、その印刷データに関して、データ展開処理を行う（ステップ S 1 2 3）。なお、このデータ展開処理は、メインの処理から起動される下位のループ処理の一例であるが、その詳細については、後に図 1 2 を用いて詳述する。このデータ展開処理からこのメインタスク処理に戻ってきた場合には、上述したステップ S 1 2 0 の処理に戻る。

【0105】

一方、ステップ S 1 2 2 において、受信したメッセージがデータ展開処理要求でないと判断した場合（ステップ S 1 2 2 : N o）には、受信したメッセージが、コマンド解析処理要求であるかどうかを判断する（ステップ S 1 2 4）。受信

したメッセージがコマンド解析処理要求である場合（ステップ S 1 2 4 : Y e s）には、そのコマンドに関して、コマンド解析処理を行う（ステップ S 1 2 5）。このコマンド解析処理の内容の説明は省略するが、このコマンド解析処理も、本実施形態におけるメインタスク処理から起動される下位の処理である。このコマンド解析処理からこのメインタスク処理に戻ってきた場合には、上述したステップ S 1 2 0 の処理に戻る。

【 0 1 0 6 】

これに対して、上述したステップ S 1 2 1 において、受信したメッセージが、リセット処理実行要求であると判断した場合（ステップ S 1 2 1 : Y e s）には、メインタスク T 3 0 は、このメインタスク T 3 0 がデータ展開処理のために R A M 1 2 に確保したメモリ領域を解放して（ステップ S 1 2 6）、データ展開処理を途中終了する。続いて、リセット処理終了通知を、ジョブコントロールタスク T 4 0 に送信する（ステップ S 1 2 7）。そして、上述したステップ S 1 2 0 の処理に戻る。

【 0 1 0 7 】

一方、上述したステップ S 1 2 4 において、受信したメッセージが、コマンド解析処理要求でないと判断した場合（ステップ S 1 2 4 : N o）には、受信したメッセージが、ジョブコントロールタスク T 4 0 から送信されたリセット処理終了通知の返信であるかどうかを判断する（ステップ S 1 2 8）。図 9 でも説明したように、ステップ S 1 2 7 でリセット処理終了通知をジョブコントロールタスク T 4 0 に送信すると、リセットするためのプリントエンジン 2 4 の機械的準備が整った段階で、ジョブコントロールタスク T 4 0 からメインタスク T 3 0 にリセット処理終了通知の返信が送信される。

【 0 1 0 8 】

受信したメッセージが、リセット処理終了通知の返信である場合（ステップ S 1 2 8 : Y e s）には、このメインタスク T 3 0 内の各種の設定状態を、初期状態に設定する（ステップ S 1 2 9）。一方、受信したメッセージが、リセット処理終了通知の返信でない場合（ステップ S 1 2 8 : N o）には、そのメッセージに対応するメインの処理を実行する（ステップ S 1 3 0）。これらステップ S 1

29及びステップS130の後、上述したステップS120の処理に戻る。

【0109】

次に、図12に基づいて、ステップS123のデータ展開処理について詳しく説明する。このデータ展開処理は、図12に示すステップS123のデータ展開処理に移行した場合の下位の処理である。したがって、このデータ展開処理が終了すると、図11のステップS120の処理に戻ることになる。

【0110】

図12に示すように、データ展開処理においては、まず、タイマーのセットを行う（ステップS140）。すなわち、タイマーの値をゼロにセットする。次に、受信バッファが空であるかどうかを判断する（ステップS141）。受信バッファが空でない場合（ステップS141：No）、すなわち、受信バッファに印刷データが格納されている場合には、その印刷データの展開を行う（ステップS142）。

【0111】

そして、このデータ展開処理を開始してから所定時間経過したかどうかを判断する（ステップS143）。具体的には、ステップS140で起動したタイマーの値に基づいて、このデータ展開処理が開始されてからの時間を取得する。そして、この取得した時間が、所定の時間（例えば、1秒）よりも大きいかどうかを判断する。

【0112】

このデータ展開処理が開始されてから所定時間経過していない場合（ステップS143：No）には、このデータ展開処理が最後まで終了したかどうかを判断する（ステップS144）。最後まで終了していない場合（ステップS144：No）には、上述したステップS141の処理に戻る。一方、最後まで終了した場合（ステップS144：Yes）には、展開し終えた印刷ジョブデータについての印刷要求を、プリントエンジン24で印刷すべく、ジョブコントロールタスクT40に送信する（ステップS145）。そして、このデータ展開処理を終了する。

【0113】

上述したステップS141で受信バッファが空であると判断した場合（ステップS141：Yes）、又は、上述したステップS143でデータ展開処理を開始してから所定時間経過していると判断した場合（ステップS143：Yes）には、メインタスクT30に、データ展開処理要求を送信し（ステップS146）、このデータ展開処理を終了する。すなわち、自分から自分に処理の起動要求を送信することにより、一旦、メインタスクT30のメインの処理に戻った後にも、再度、このデータ展開処理が起動されるようにする。

【0114】

次に、図13に基づいて、ジョブコントロールタスクT40で行われるジョブコントロール処理について説明する。本実施形態においても、このジョブコントロール処理は、ROM14に格納されているジョブコントロール処理プログラムを、リアルタイムOS上でCPU16が実行することにより実現される。

【0115】

図13に示すように、ジョブコントロールタスクT40は、まず、何からのメッセージを受信したかどうかを判断する（ステップS170）。何らメッセージを受信していない場合（ステップS170：No）には、このステップS170の処理を繰り返して待機する。

【0116】

何らかのメッセージを受信した場合（ステップS170：Yes）には、リセット中フラグRSFがセットされているかどうかを判断する（ステップS171）。リセット中フラグRSFがセットされていない場合（ステップS171：No）には、受信したメッセージが、制御コマンド解釈タスクT20から送信されたリセットコマンド受信のメッセージであるかどうかを判断する（ステップS172）。

【0117】

受信したメッセージがリセットコマンド受信のメッセージでない場合（ステップS172：No）には、そのメッセージについて必要なジョブを実行し、そのコントロールを行う（ステップS173）。そして、上述したステップS170に戻る。

【0118】

一方、上述したステップS172で、受信したメッセージがリセットコマンド受信のメッセージである場合（ステップS172：Yes）には、ジョブコントロールタスクT40は、リセット中フラグRSFをセットする（ステップS174）。続いて、ジョブコントロールタスクT40は、メインタスクT30へ、リセット処理実行要求を送信する（ステップS175）。そして、上述したステップS170に戻る。

【0119】

一方、上述したステップS171の処理で、リセット中フラグRSFがセットされていると判断した場合（ステップS171：Yes）には、受信したメッセージが、何らかの応答を必要としているメッセージであるかどうかを判断する（ステップS176）。応答が必要なメッセージでない場合（ステップS176：No）には、そのメッセージを読み捨てる（ステップS177）。メッセージを読み捨てることにより、ジョブが読み捨てられたことになる。そして、上述したステップS170の処理に戻る。

【0120】

一方、上述したステップS176で、受信したメッセージが応答の必要なメッセージである場合（ステップS176：Yes）には、ジョブコントロールタスクT40は、そのメッセージが、メインタスクT30からのリセット処理終了通知であるかどうかを判断する（ステップS178）。受信したメッセージが、リセット処理終了通知でない場合（ステップS178：No）には、メインタスクT30がまだ最上位のループ処理に戻ってきていないことになるので、とりあえず、受信したメッセージに対して必要な返事を送信しておく（ステップS179）。これは、メインタスクT30の下位のループ処理で、ジョブコントロールタスクT40からの返信待ちになっているものがある可能性があるため、このループ処理を進行させるために、必要な返事のみは返信する必要があるからである。但し、ジョブコントロールタスクT40では、そのメッセージの内容は実行せず、読み捨てる。そして、上述したステップS170の処理に戻る。

【0121】

一方、上述したステップS178で、受信したメッセージが、メインタスクT30からのリセット処理終了通知である場合（ステップS178：Yes）には、メインタスクT30が最上位のループに戻って、確保していたメモリ領域を解放したことになるので、ジョブコントロールタスクT40は、エンジン制御タスクT50に、機械的状態確認メッセージを送信する（ステップS180）。この機械的状態確認メッセージを受信したエンジン制御タスクT50は、プリントエンジン24の機械的な状態を初期状態にリセットした後、機械的状態初期化終了メッセージをジョブコントロールタスクT40に送信する。

【0122】

このため、ジョブコントロールタスクT40では、ステップS180で機械的状態確認メッセージを送信した後は、この機械的状態初期化終了メッセージを受信するまで待機する（ステップS181）。この機械的状態初期化終了メッセージを受信した場合（ステップS181：Yes）には、ジョブコントロールタスクT40は、メインタスクT30に、リセット処理終了通知の返事を送信する（ステップS182）。

【0123】

次に、ジョブコントロールタスクT40は、これでプリンタ部のリセットが終了したので、リセット中フラグRSFをリセットする（ステップS183）。具体的には、リセット中フラグRSFを「0」にする。そして、上述したステップS170の処理に戻る。

【0124】

以上のように、本実施形態に係るマルチファンクションプリンタ10によれば、スキャナ部とプリンタ部とが並行して動作している場合でも、プリンタ部のみをリセットすることができるようにしたので、ユーザの利便性が向上する。すなわち、スキャナ部とプリンタ部とが並行して動作している場合に、プリンタ部をリセットしようとして、スキャナ部のスキャン動作までもが停止してしまう事態を回避することができる。

【0125】

具体的には、プリンタ部リセットコマンドを受信した場合には、メインタスク

T30に、リセット処理実行要求を送信する。このリセット処理実行要求を受信したメインタスクT30は、データ展開処理のために確保したメモリ領域を解放して、データ展開処理を途中終了する。また、このメインタスクT30のメインの処理から下位のループ処理が起動されている場合でも、この下位のループ処理は、その処理の途中であっても、所定時間経過後にメインの処理に一旦戻ってくるように設定されている。このため、メインタスクT30の処理が下位のループ処理を実行している場合でも、リセット処理実行要求をメインタスクT30が受信した場合には、そのデータ展開処理を終了させることができる。

【0126】

なお、本発明は上記実施形態に限定されず種々に変形可能である。例えば、上述した第1実施形態における図5のデータ展開制御処理、及び、図6のグラフィックマネージャ処理では、ステップS43及びステップS62で入力バッファから読み込んだデータが0x00であるかどうかの判断をしてから、リセット中フラグRSFがセットされているかどうかを判断したが、これらステップS43及びステップS62の処理を省略して、直接、リセット中フラグRSFがセットされているかどうかを判断するようにしてもよい。

【0127】

また、上述した第1及び第2実施形態における各種のタスクの処理区分は上述したものに限定されるものではなく、複数のタスクの処理をまとめて1つのタスクとして構成してもよく、また、1つのタスクの処理を複数のタスクに分割して構成してもよい。

【0128】

さらに、上述した第1実施形態におけるインターフェース制御タスクT10では、リセット中フラグRSFとは別に、プリンタ部がリセット中であることを示すリセット処理中フラグWSFを設けることとしたが、このリセット処理中フラグWSFを別途設けずにリセット中フラグRSFを共用するようにしてもよい。

【0129】

また、上述した第2実施形態においては、メインタスクT30は、ジョブコントロールタスクT40を介して、プリンタエンジン24を初期化するためのメッ

セージをエンジン制御タスクT50に送信することとしたが、このメッセージを直接送信するようにしてもよい。すなわち、機械的状态確認メッセージをメインタスクT30からエンジン制御タスクT50に送信し、エンジン制御タスクT50がプリントエンジン24の機械的状态の初期化を行った後に、機械的状态初期化終了メッセージをエンジン制御タスクT50からメインタスクT30に送信するようにしてもよい。

【0130】

さらに、上述した第1実施形態及び第2実施形態においては、コンピュータからマルチファンクションプリンタ10にプリンタ部リセットコマンドを送信することとしたが、マルチファンクションプリンタ10におけるプリンタ部のリセットの仕方は、この態様に限られるものではない。例えば、マルチファンクションプリンタ10の本体にリセットボタンを設け、ユーザがこのリセットボタンを操作したときに、プリンタ部リセットコマンドがマルチファンクションプリンタ10で生成され、インターフェース制御タスクT10に入力されるようにしてもよい。

【0131】

また、上述の実施形態で説明したインターフェース制御タスクT10、制御コマンド解釈タスクT20、メインタスクT30、ジョブコントロールタスクT40、及び、エンジン制御タスクT50等の各タスク処理については、これら各処理を実行するためのプログラムをフロッピーディスク、CD-ROM (Compact Disc-Read Only Memory)、ROM、メモリカード等の記録媒体に記録して、記録媒体の形で頒布することが可能である。この場合、このプログラムが記録された記録媒体をマルチファンクションプリンタ10に読み込ませ、実行させることにより、上述した実施形態を実現することができる。

【0132】

また、マルチファンクションプリンタ10は、オペレーティングシステムや別のアプリケーションプログラム等の他のプログラムを備える場合がある。この場合、マルチファンクションプリンタ10の備える他のプログラムを活用し、記録媒体にはそのマルチファンクションプリンタ10が備えるプログラムの中から、

上述した実施形態と同等の処理を実現するプログラムを呼び出すような命令を記録するようにしてもよい。

【0133】

さらに、このようなプログラムは、記録媒体の形ではなく、ネットワークを通じて搬送波として頒布することも可能である。ネットワーク上を搬送波の形で伝送されたプログラムは、マルチファンクションプリンタ10に取り込まれて、このプログラムを実行することにより上述した実施形態を実現することができる。

【0134】

また、記録媒体にプログラムを記録する際や、ネットワーク上を搬送波として伝送される際に、プログラムの暗号化や圧縮化がなされている場合がある。この場合には、これら記録媒体や搬送波からプログラムを読み込んだマルチファンクションプリンタ10は、そのプログラムの復号化や伸張化を行った上で、実行する必要がある。

【0135】

【発明の効果】

以上説明したように、本発明によれば、スキャナ部とプリンタ部とを有するマルチファンクションプリンタにおいて、スキャナ部が動作中であっても、プリンタ部だけをリセットすることができるようにしたので、ユーザの利便性が向上する。

【図面の簡単な説明】

【図1】

本発明の一実施形態に係るマルチファンクションプリンタの内部構成を示すブロック図である。

【図2】

第1実施形態に係るマルチファンクションプリンタがコンピュータからプリンタ部リセットコマンドを受信した場合の処理を概略的に説明するタイミングチャートを示す図である。

【図3】

本実施形態に係るインターフェース制御処理（インターフェース制御タスク）

の内容を説明するフローチャートである。

【図4】

本実施形態に係る制御コマンド解釈処理（制御コマンド解釈タスク）の内容を説明するフローチャートである。

【図5】

本実施形態に係るデータ展開制御処理（メインタスク）の内容を説明するフローチャートである。

【図6】

本実施形態に係るグラフィックマネージャ処理（メインタスク）の内容を説明するフローチャートである。

【図7】

本実施形態に係るジョブコントロール処理（ジョブコントロールタスク）の内容を説明するフローチャートである。

【図8】

本実施形態に係るネゴシエーション処理の内容を説明するフローチャートである。

【図9】

第2実施形態に係るマルチファンクションプリンタがコンピュータからプリンタ部リセットコマンドを受信した場合の処理を概略的に説明するタイミングチャートを示す図である。

【図10】

第2実施形態に係るインターフェース制御処理（インターフェース制御タスク）の内容を説明するフローチャートである。

【図11】

第2実施形態に係るメインタスク処理（メインタスク）の内容を説明するフローチャートである。

【図12】

第2実施形態に係るデータ展開処理（メインタスク）の内容を説明するフローチャートである。

【図 1 3】

第 2 実施形態に係るジョブコントロール処理（ジョブコントロールタスク）の内容を説明するフローチャートである。

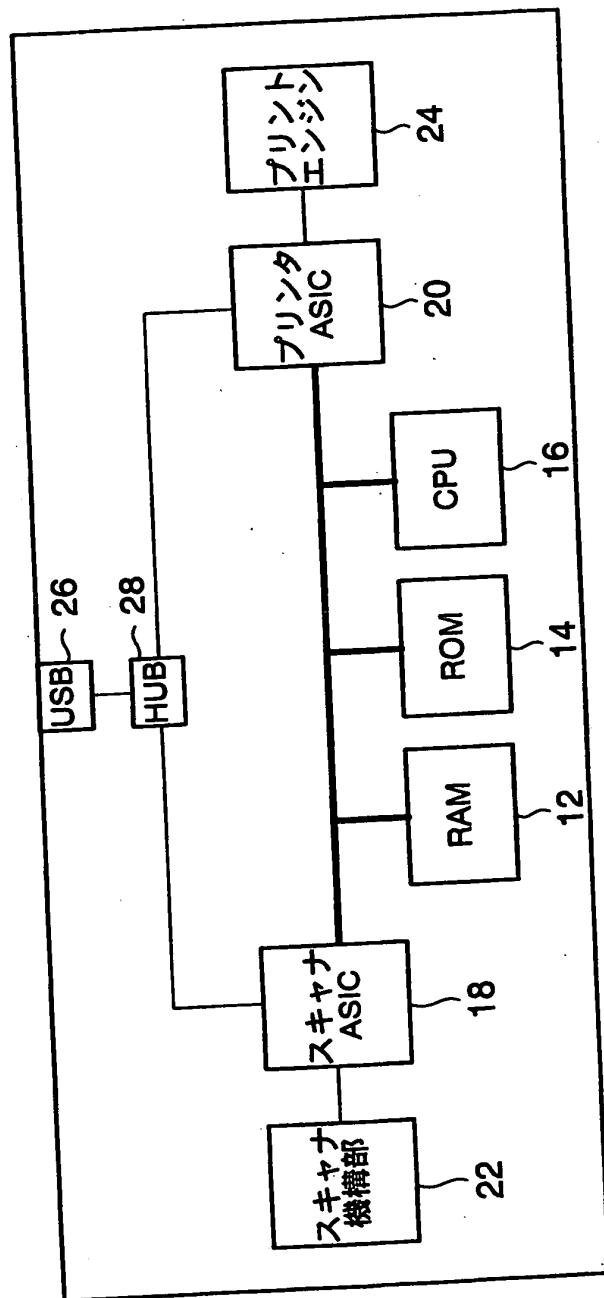
【符号の説明】

- 1 0 マルチファンクションプリンタ
- 1 2 R A M
- 1 4 R O M
- 1 6 C P U
- 1 8 スキャナ A S I C
- 2 0 プリンタ A S I C
- 2 2 スキャナ機構部
- 2 4 プリントエンジン
- 2 6 U S B ポート
- 2 8 U S B ハブ
- T 1 0 インターフェース制御タスク
- T 2 0 制御コマンド解釈タスク
- T 3 0 メインタスク
- T 4 0 ジョブコントロールタスク

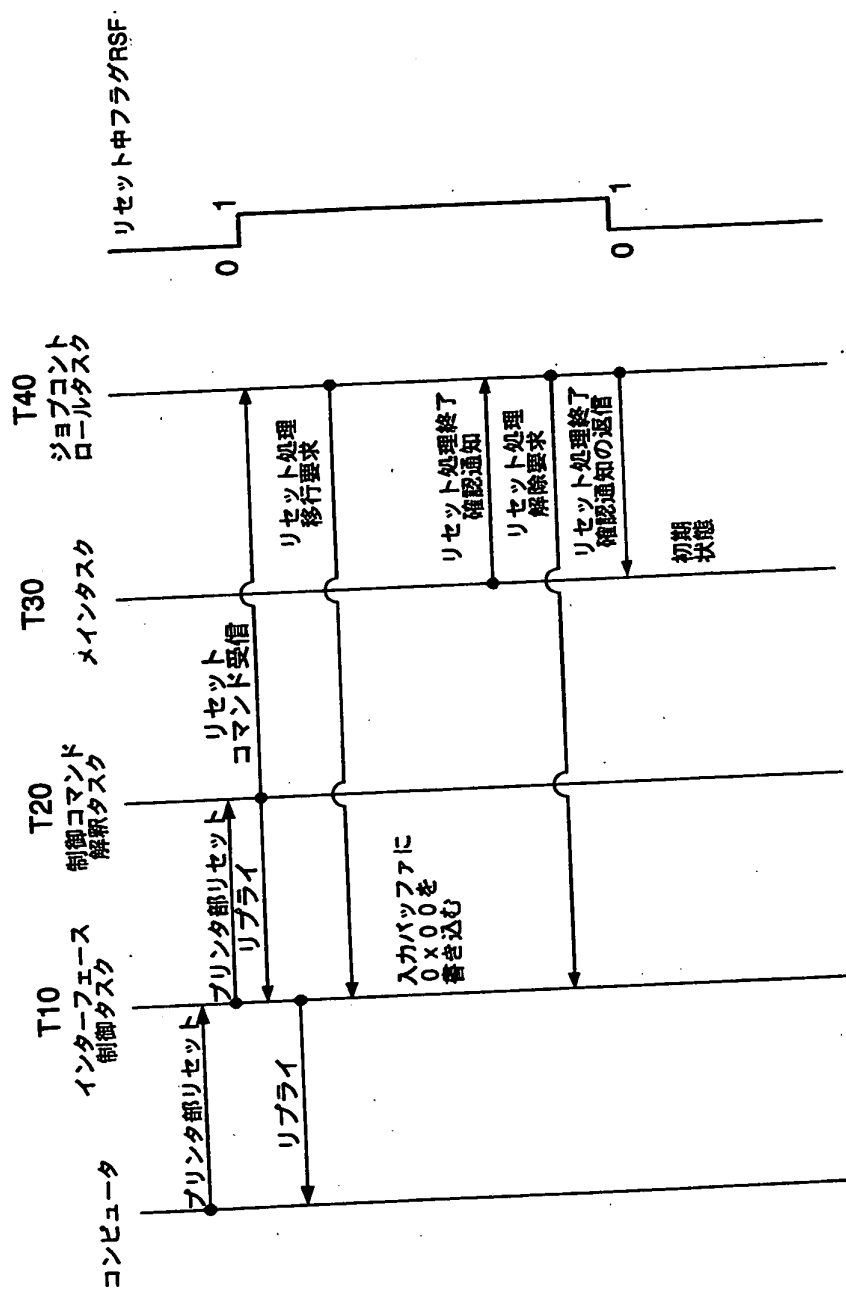
【書類名】 図面

【図1】

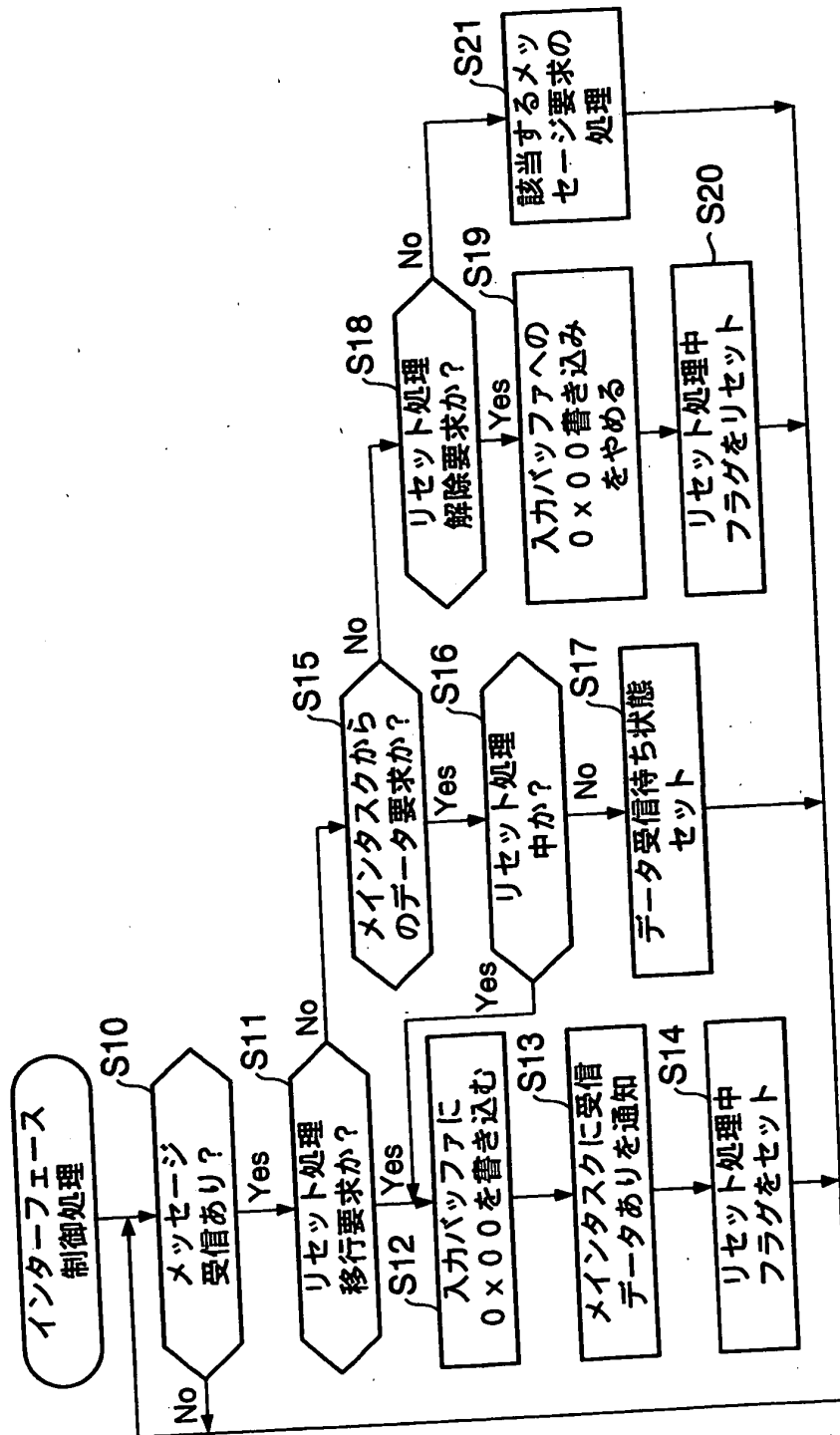
10:マルチファンクションプリンタ



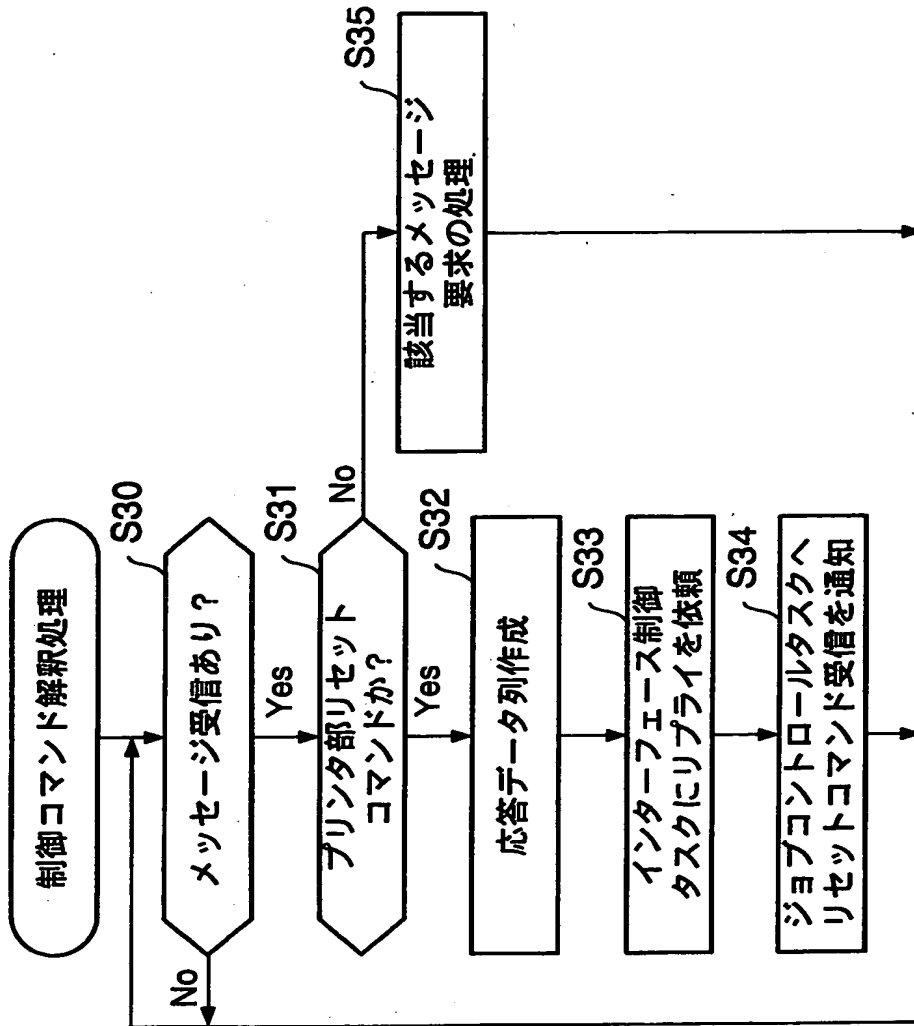
【図2】



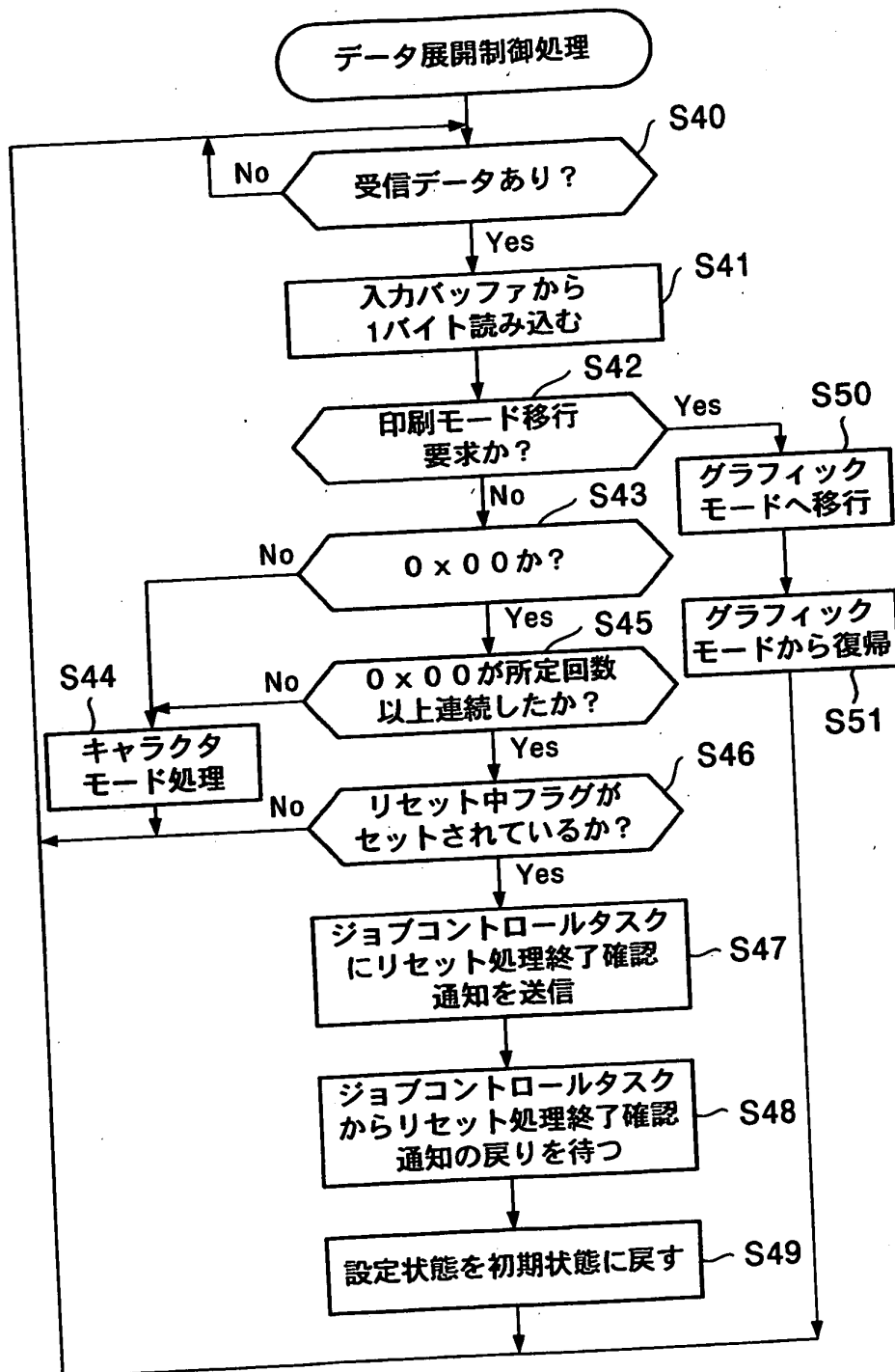
【図3】



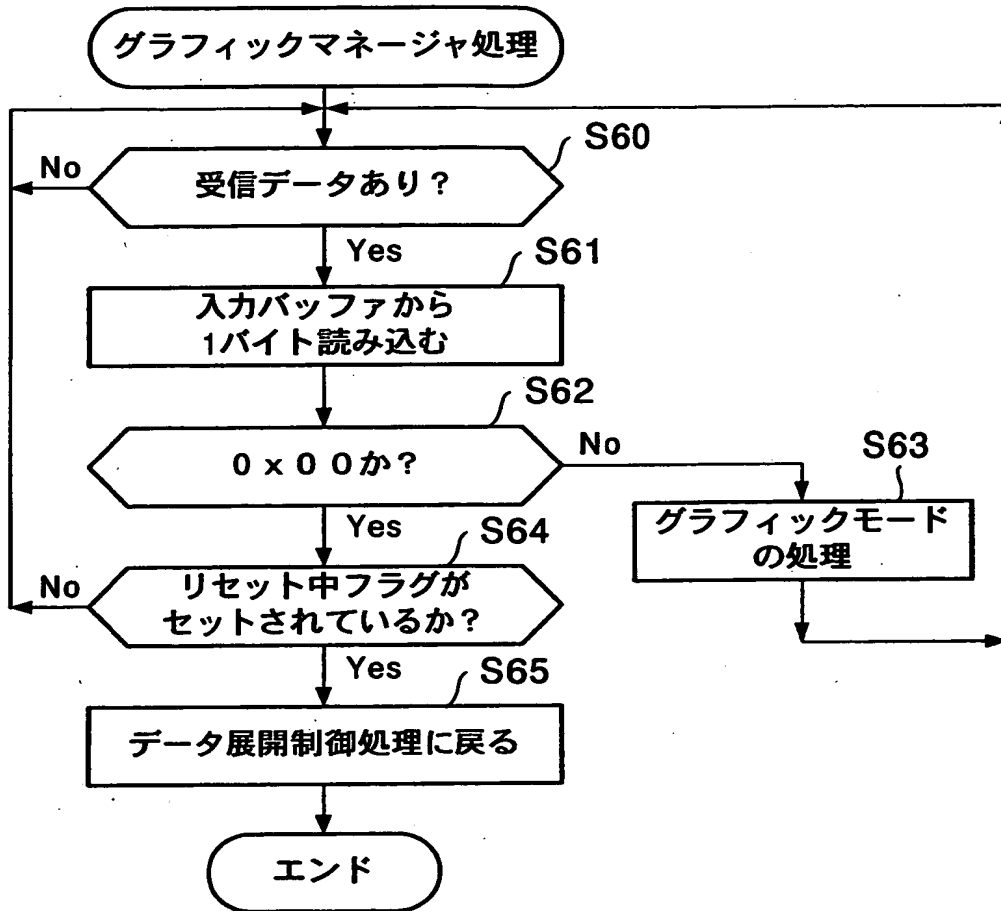
【図 4】



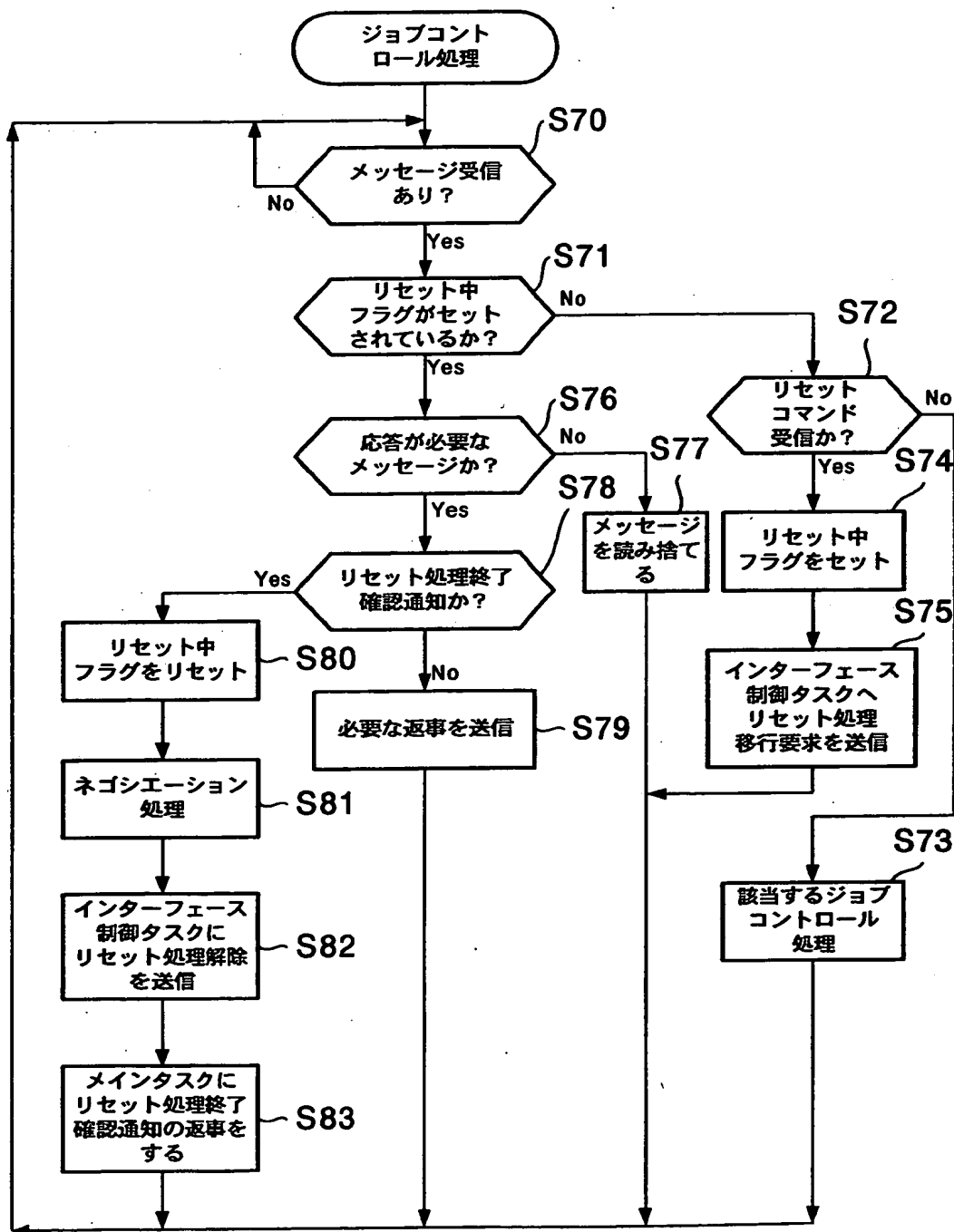
【図 5】



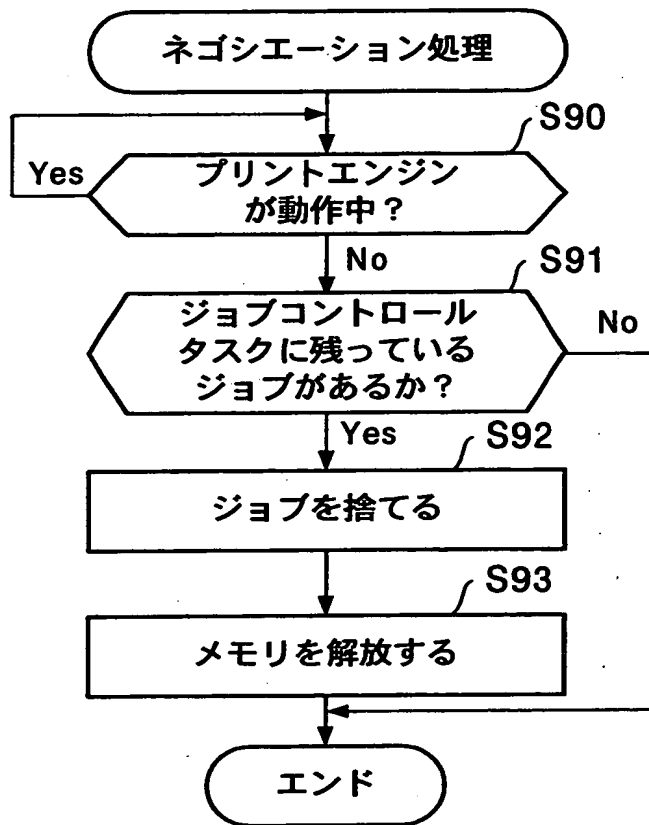
【図 6】



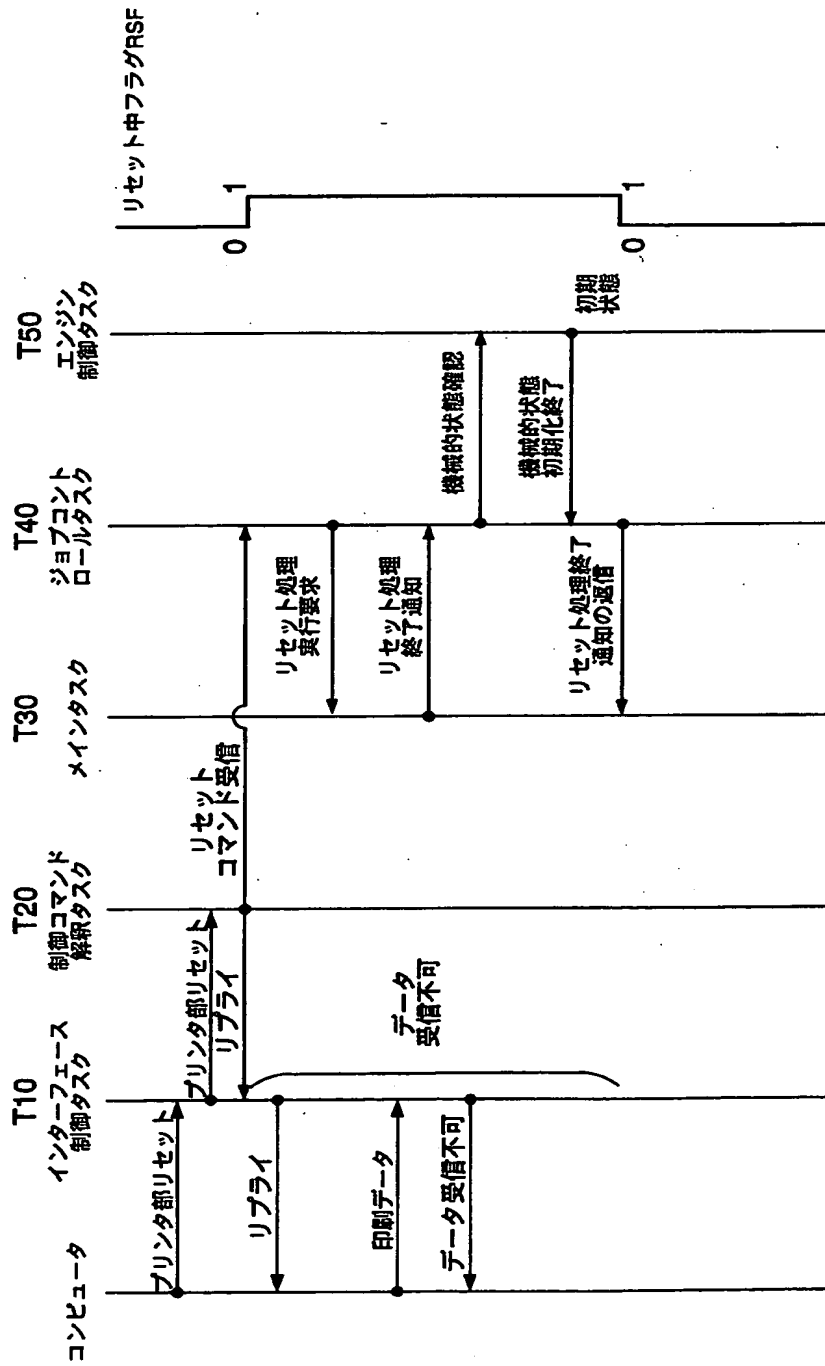
【図 7】



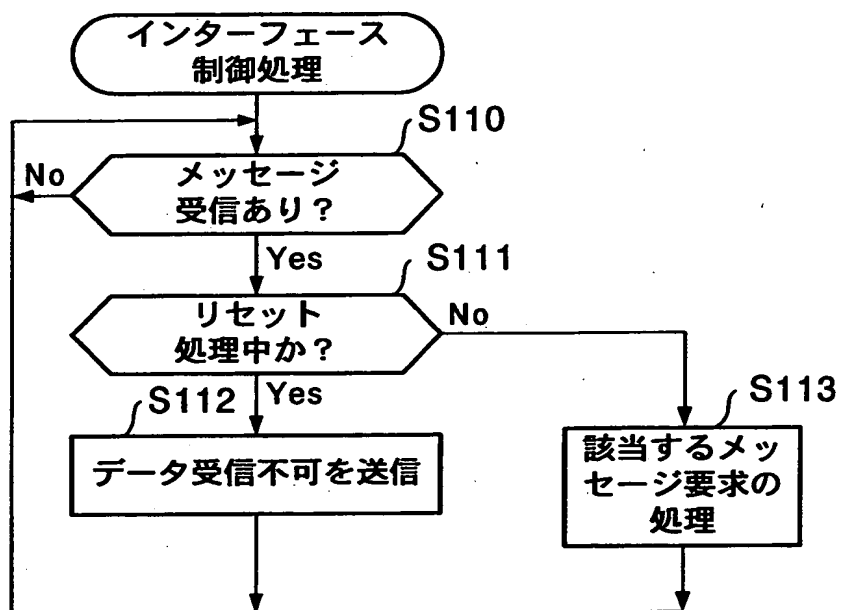
【図 8】



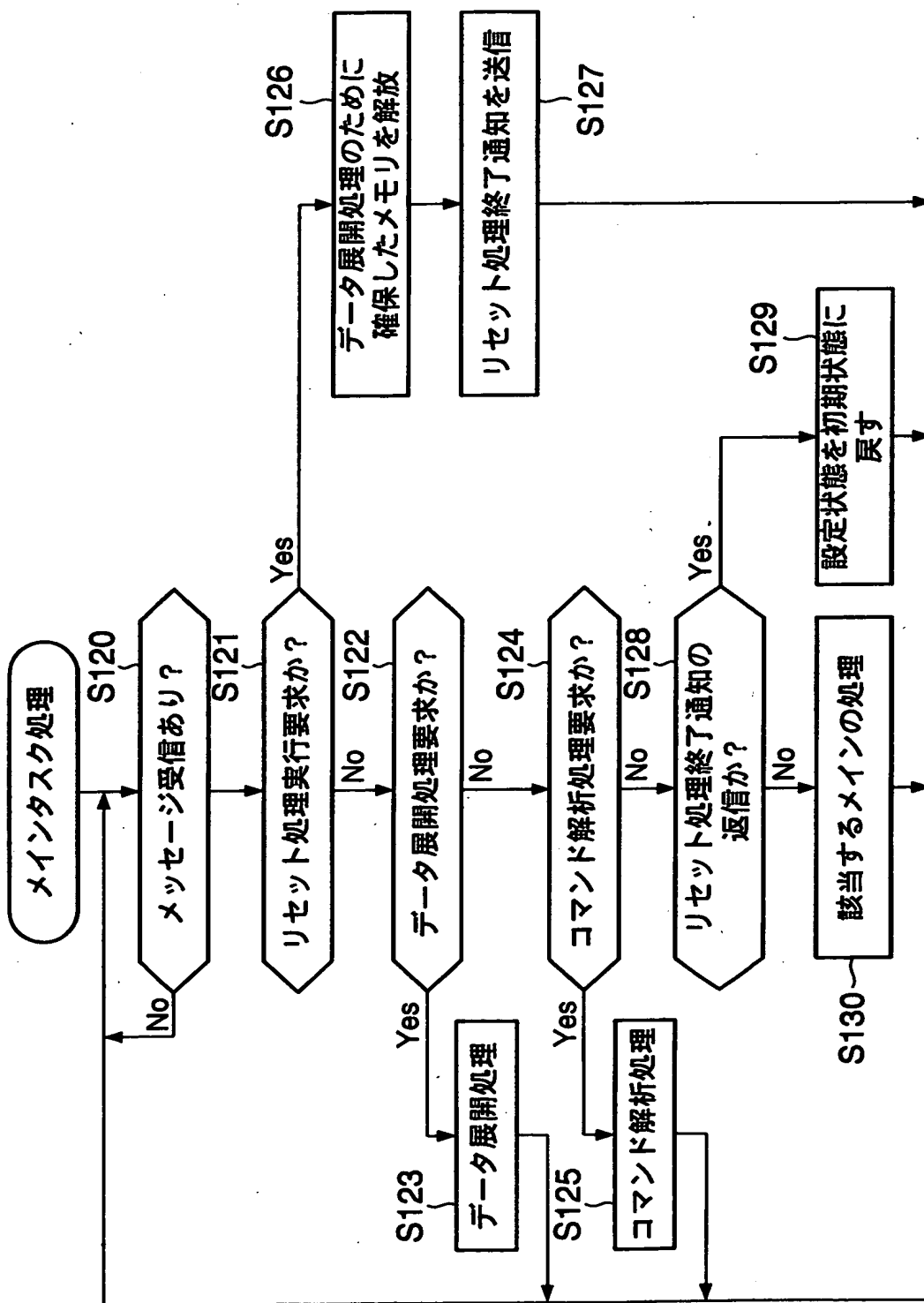
【図9】



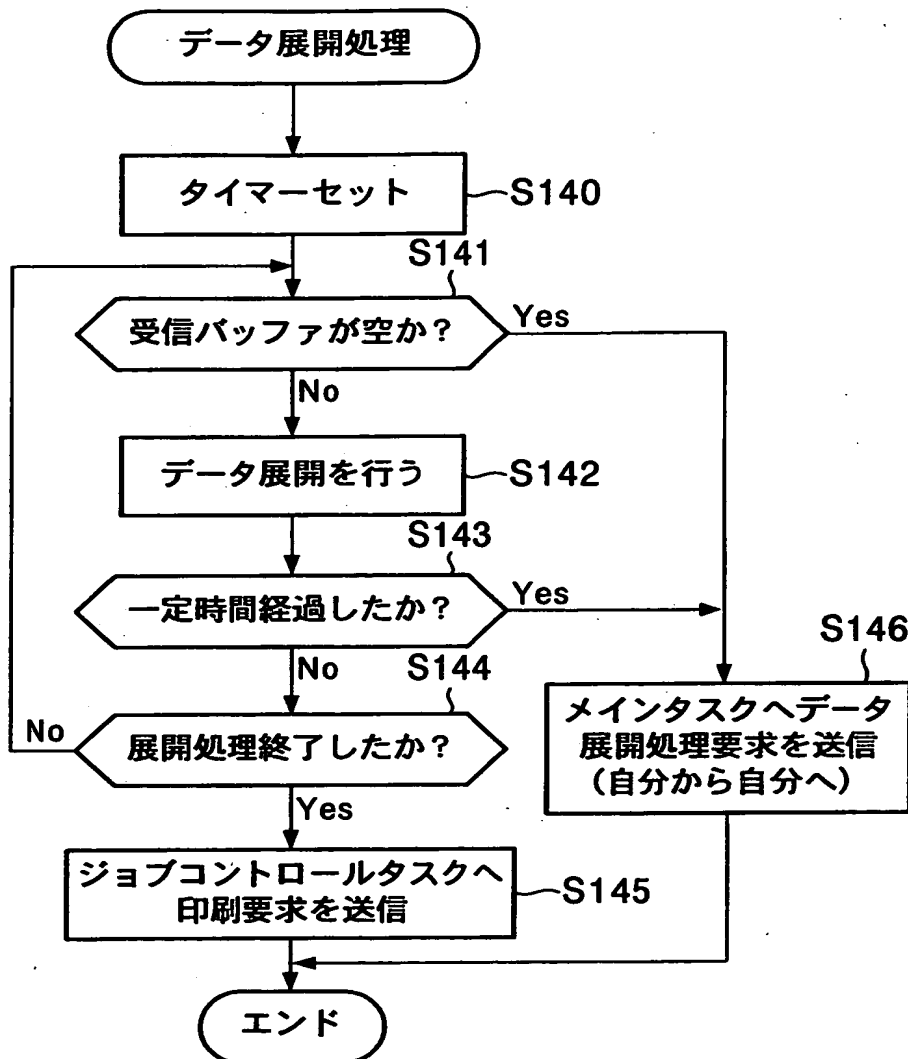
【図 10】



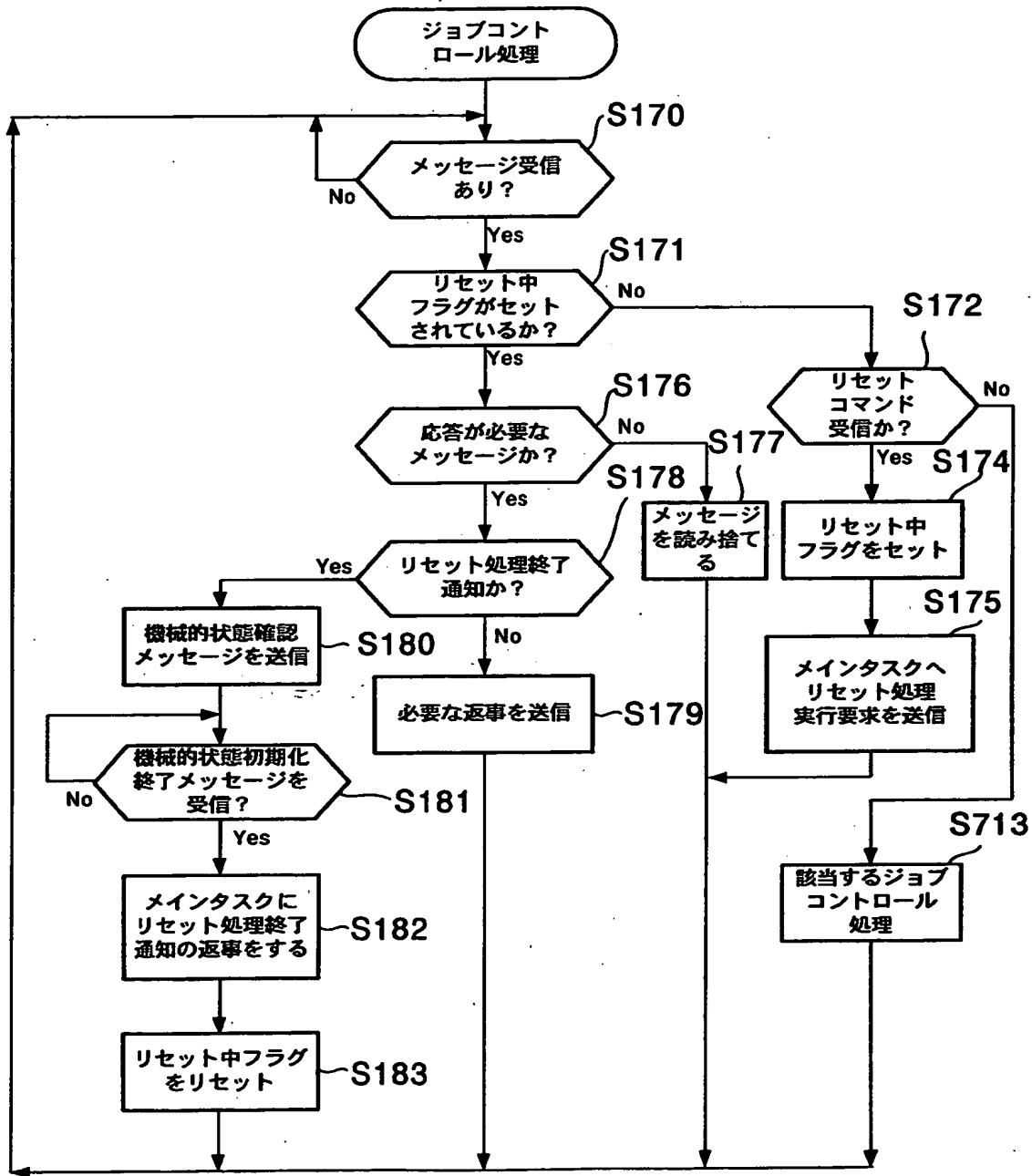
【図 11】



【図12】



【図13】



【書類名】 要約書

【要約】

【課題】 スキャナ部とプリンタ部とを有するマルチファンクションプリンタにおいて、プリンタ部のリセットのみを可能にする。

【解決手段】 スキャナとプリンタとが一体化されたマルチファンクションプリンタ10において、プリンタ部リセットコマンドを受信した場合には、インターフェース制御タスクT10は、入力バッファにNULLを強制的に書き込むようにし、ジョブコントロールタスクT40は、リセット中フラグRSFをセットするようにする。メインタスクT30では、階層構造の各種ループ処理が実行されているが、入力バッファにNULLが格納されているのでやがてその処理を終了し、リセット中フラグRSFがセットされているので、メインループ処理に次第に戻ってくる。このため、スキャナ部とプリンタ部とが並行して動作している場合でも、メインタスクT30を初期化でき、プリンタ部のみをリセットすることができるようになる。

【選択図】 図1

認定・付加情報

特許出願の番号	特願2001-381409
受付番号	50101837230
書類名	特許願
担当官	第七担当上席 0096
作成日	平成13年12月19日

<認定情報・付加情報>

【特許出願人】

【識別番号】	000002369
【住所又は居所】	東京都新宿区西新宿2丁目4番1号
【氏名又は名称】	セイコーエプソン株式会社

【代理人】

申請人

【識別番号】	100075812
--------	-----------

【住所又は居所】	東京都千代田区丸の内3-2-3 協和特許法律事務所
----------	---------------------------

【氏名又は名称】	吉武 賢次
----------	-------

【選任した代理人】

【識別番号】	100088889
--------	-----------

【住所又は居所】	東京都千代田区丸の内3丁目2番3号 協和特許法律事務所
----------	-----------------------------

【氏名又は名称】	橘谷 英俊
----------	-------

【選任した代理人】

【識別番号】	100082991
--------	-----------

【住所又は居所】	東京都千代田区丸の内3丁目2番3号 富士ビル 協和特許法律事務所
----------	-------------------------------------

【氏名又は名称】	佐藤 泰和
----------	-------

【選任した代理人】

【識別番号】	100096921
--------	-----------

【住所又は居所】	東京都千代田区丸の内3-2-3 富士ビル3階 協和特許法律事務所
----------	-------------------------------------

【氏名又は名称】	吉元 弘
----------	------

【選任した代理人】

【識別番号】	100103263
--------	-----------

【住所又は居所】	東京都千代田区丸の内3丁目2番3号 協和特許法律事務所
----------	-----------------------------

次頁有

認定・付加情報（続き）

【氏名又は名称】	川崎 康
【選任した代理人】	
【識別番号】	100107582
【住所又は居所】	東京都千代田区丸の内3丁目2番3号 協和特許 法律事務所
【氏名又は名称】	関根 毅

特2001-381409

出 願 人 履 歴 情 報

識別番号 [000002369]

1. 変更年月日	1990年 8月20日
[変更理由]	新規登録
住 所	東京都新宿区西新宿2丁目4番1号
氏 名	セイコーエプソン株式会社